



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

**FACULTAD DE INGENIERÍA**

**Sistema Administrador de la  
Producción en Empresa de  
Artes Gráficas SiRPAG**

**TESIS**

Que para obtener el título de  
**Ingeniera en Computación**

**P R E S E N T A**

Ana Laura Gutiérrez Miranda

**DIRECTOR DE TESIS**

Ing. Orlando Zaldívar Zamorategui



**Ciudad Universitaria, Cd. Mx., 2018**

## Índice Temático

Agradecimiento.....	II
Capítulo 1 Introducción .....	1
1.1. Planteamiento del problema.....	2
1.2. Objetivo general.....	3
1.3. Objetivo específico.....	3
Capítulo 2 Marco teórico .....	4
2.1. Introducción .....	5
2.2. Sistema de información.....	5
2.3. Ingeniería de software.....	6
2.3.1. Modelos del proceso del software.....	7
2.4. Metodología en cascada modificada.....	8
2.4.1. Análisis y definición de los requerimientos .....	9
2.4.2. Diseño .....	11
2.4.2.1. Bases de datos .....	12
2.4.2.2. Interfaz .....	14
2.4.3. Implementación con Visual FoxPro .....	17
2.4.4. Integración.....	18
2.4.5. Funcionamiento y mantenimiento.....	18
2.4.5.1. Dificultades en el mantenimiento.....	20
2.4.6. Pruebas.....	21
Capítulo 3 Caso práctico.....	23
3.1. Inicio del proyecto: especificación de requerimientos de software.....	24
3.1.1. Introducción.....	24
3.1.2. Requerimientos del cliente.....	25
3.1.3. Requerimientos del desarrollador .....	29
3.2. Diseño del sistema.....	37
3.2.1. Diseño preliminar.....	37
3.2.2. Diseño detallado.....	41
3.2.3. Jerarquía de control.....	61
3.2.4. Procedimiento del software.....	62
3.2.5. Diseño de la interfaz.....	75
3.2.6. Calidad del diseño .....	96
3.3. Implementación .....	98
3.4. Integración.....	105
3.5. Funcionamiento .....	107
Capítulo 4 Resultados, impacto y conclusiones .....	110
4.1. Resultados.....	111
4.2. Impacto en la empresa.....	122
4.3. Conclusiones .....	127
Glosario de términos .....	129
Bibliografía.....	131

## Agradecimiento

---

Primeramente, agradezco a la Facultad de Ingeniería de la Universidad Nacional Autónoma de México por los años de educación y formación académica que recibí dentro y fuera de sus instalaciones. Es un honor haber estudiado en esta máxima casa de estudios, una experiencia que me llena de orgullo y satisfacción, que nunca olvidaré. Las enseñanzas que obtuve por parte de los profesores y mi asesor de tesis el Ing. Orlando Zaldívar Zamorategui, los libros que complementaron mi aprendizaje y las experiencias personales que me han formado hasta el día de hoy, por todo esto y más es que con gran entusiasmo puedo decir que he tenido una de las mejores oportunidades de mi vida al estudiar en la UNAM.

Agradezco a Dios por brindarme la salud, sabiduría y conocimiento para salir adelante a pesar de cada obstáculo y tropiezo, transformándolos en aprendizaje. Que mi corazón nunca olvide todos los momentos en los cuales encontré consuelo en su Palabra y renovaron mis fuerzas para dar siempre un paso adelante.

Asimismo, a mi familia, por su paciencia, amor y disciplina que me forjaron el carácter ante las circunstancias adversas, la calma para obtener los frutos que en su tiempo llegarían, los años de dedicación a mi salud, educación y valores. No tengo forma de agradecer todo lo que han hecho por mí hasta el día de hoy. Espero poder siempre corresponder al infinito amor que me han dado.

A mi novio Iván Dávila por contagiarme siempre de su paciencia para ser fuerte ante las injusticias y mostrar la mejor actitud en la tormenta. Por último, a mis seres queridos que conocí a lo largo de mi vida personal y académica, que ya sea en pequeña o grande medida me brindaron aprendizaje que llevaré conmigo siempre.

# **Capítulo 1**

# **Introducción**

En la actualidad, los sistemas de información han evolucionado a una velocidad impresionante, sin importar cuán grande pueda ser la resistencia al cambio, las empresas han tenido que aceptar esta realidad y ser parte de ella para continuar creciendo y triunfando en este siglo que se ha identificado por sus grandes avances tecnológicos.

Un sistema que recolecta información de la producción de una compañía de artes e impresión es una herramienta indispensable para conocer los niveles de producción y áreas de oportunidad en las distintas áreas que componen a la organización. El objetivo de un Sistema Administrador de la Producción es recolectar la información real de las diferentes actividades realizadas por los operadores que afecten de manera directa al rendimiento de la producción y trabajo de las máquinas, después se almacene calculando de manera automática el tiempo que le demoró al operador realizar su trabajo y otorga la opción de consultar y descargar los resultados.

Así, el Sistema Administrador de la Producción en Empresa de Artes Gráficas (SiRPAG) busca brindar un servicio y apoyo a las distintas áreas, descubriendo áreas de oportunidad que puedan ser abordadas y definir metas para optimizar el rendimiento y la productividad de la compañía. El conocimiento es poder, y un Sistema de Información (SI) entrega datos confiables a la organización de lo que ocurre en el área productiva sin tener que realizar un recorrido o estudio de campo.

No existe un SI genérico y compatible para todas las organizaciones del mundo, se tiene que desarrollar desde los cimientos considerando cada una de las necesidades específicas de cada una de sus áreas que la componen.

El presente trabajo busca documentar los procesos implementados para el desarrollo de SiRPAG, utilizando la metodología en cascada modificada, con el propósito de cubrir las necesidades de recolección, proceso y consulta de información de la producción de la Compañía Mexicana de Impresión de Artes gráficas Excelform.

En el **Capítulo 1** se plantean los antecedentes y problema que presenta la Compañía y la solución que entregará este proyecto. En el **Capítulo 2** se exponen cada uno de los elementos que se requieren para el desarrollo del SI que cumplirá con los requerimientos y necesidades de la organización, cada punto que explica será usado más adelante de manera práctica, exponiendo lo que es un SI, cómo se desarrolla y los componentes que se requieren haciendo uso de una metodología formal de desarrollo de sistemas como lo es la metodología en cascada modificada. El **Capítulo 3** es el desarrollo de SiRPAG y los elementos explicados en el capítulo anterior de manera detallada con diagramas y tablas. Por último, el **Capítulo 4** expone los resultados obtenidos, el cumplimiento de las necesidades, su impacto sobre la empresa y conclusiones sobre los beneficios de implementar la metodología.

## 1.1. Planteamiento del problema

Las medianas empresas dedicadas al área de artes gráficas e impresión variable corren un riesgo que, a la medida de ir consiguiendo nuevos clientes que requieren diferentes tipos de productos como: posters, flyers, impresión de contratos, cartas cobranza, estados de cuenta, impresión en vinil, etc. la demanda de trabajo hacia las máquinas es más recurrente y por consecuente son inevitables los cuellos de botella en la fase de producción.

Por lo que es necesario optar por cambios en las actividades para disminuir este percance, pero al ser una empresa grande, unas cuantas personas no serían capaces de registrar los tiempos de operación de cada una de las actividades de todos los trabajadores de la planta y capturarlas en la computadora.

Tener los tiempos de producción al momento que se requieran es lo principal que se necesitaría para iniciar con el análisis y crear una contramedida, por lo que sí, colocando equipos de cómputo en cada una de las áreas y que el empleado al teclear el número de orden de trabajo del pedido que están laborando y al término volver a escribir dicho número en un sistema, se podrían obtener los tiempos de una forma más rápida y generar informes en todas las áreas y así seguir el estatus de producción de cualquier pedido y consultar en cualquier momento. Todo esto se realiza en pasos que no toman más de un minuto para agilizar el registro, sin tener que redactar cada actividad en reportes escritos a papel que toma tiempo, tanto para el usuario como para el capturista que ingresa los datos requeridos en la computadora.

## 1.2. Objetivo general

Elaboración de un sistema administrador de la producción en lenguaje de programación Visual FoxPro para una empresa que labora en el ramo de las artes gráficas y dato variable, cuyo propósito es apoyar a las diferentes áreas que componen dicha empresa y así gestionar y organizar de una manera efectiva las horas de trabajo tanto de máquina como operador y obtener tiempos óptimos de producción. Para la creación del sistema SiRPAG se aplicará la metodología en cascada modificada.

## 1.3. Objetivo específico

Diseño y programación de un sistema (con la estructura de un formulario) cuya finalidad sea que, al ingresar datos estructurados por parte del operador, éstos sean almacenados y posteriormente procesados para cumplir con su objetivo: proporcionar información detallada y organizada que ayude a obtener un control de los tiempos de trabajo por máquina, producto y operador.

La empresa cuenta con más de 60 trabajadores que operan alrededor de 40 máquinas, es necesario constantemente entregar los resultados de tiempo por operación y hallar una alternativa para reducir los cuellos de botella a la hora de cumplir con algún pedido.

Este sistema está orientado a apoyar al Área de Calidad y los responsables de las diferentes áreas de trabajo que componen a la Empresa: Prensa Plana, Impresión Digital, Diseño, Dato Variable, Formas Continuas, etc. Desde el ingreso de un pedido, hasta la actualización de las actividades en que se encuentren trabajando, edición, consulta y monitoreo; cada uno de los puntos que son necesarios conocer del estatus de una actividad de producción y para poder abarcar las necesidades de la empresa en calidad y tiempo de trabajo, todo esto almacenado en una base de datos, la cual será actualizada por un sistema de información con una interfaz de usuario similar al de un formulario, pero que cuente con la característica de hacer consultas específicas de tiempos de operación de las máquinas y operadores.

# **Capítulo 2**

# **Marco**

# **teórico**

## 2.1. Introducción

En este capítulo se verán los elementos requeridos para el desarrollo e implementación de SiRPAG, desde el concepto de un SI, metodología de desarrollo, sus componentes, etc. Cada tema explicado cumplirá una función indispensable en este proyecto.

## 2.2. Sistema de información

Los SI se encargan de recopilar, procesar y administrar un conjunto de datos para ayudar a tomar decisiones, controlar operaciones, analizar problemas y crear nuevos productos o servicios a partir de esta información. Todo SI nace, se desarrolla y muere con la empresa hablando de manera general. Los SI pueden ser desde lo más simple en su diseño, proceso y evolucionar o ser remplazados por uno más robusto, pero siempre serán componentes esenciales en una organización mientras exista, hoy día la computación ha demostrado ser el medio seguro para almacenamiento, proceso y consulta de la información.

Hay 3 acciones que se realizan en un SI:

- *Entrada*: captura de los datos de interés para el usuario final.
- *Proceso*: tras la captura de los datos, los convierte en algo más significativo. Porque un dato es un valor específico y simple, pero el conjunto de estos datos relacionados y agrupados se convierten en un registro que dará al usuario algún tipo de información.
- *Salida*: transfiere la información procesada al interesado para usarla en las actividades que la involucran.

El uso que hoy día se da a los SI en una empresa es para el apoyo de los procesos de negocios y funciones empresariales que a diferencia de cualquier otro software almacena gran cantidad de información útil para tomar decisiones dentro de la organización.

El SI se compone de seis elementos fundamentales, que son:

- *Base de datos*: lugar donde se almacenará la información recaudada por el sistema, la cual se organiza en registros identificables por una clave primaria.
- *Transacciones*: se refiere a todos los componentes de la interfaz que permiten la comunicación con el usuario para agregar, consultar, modificar o eliminar un registro.
- *Informes*: componentes que otorgan al usuario información a partir de un criterio de búsqueda.
- *Procesos*: elementos de control que permiten al usuario proveer información en la base de datos y generen nuevos registros. Ejemplo: Al dar clic en "Finalizar tarea" el sistema guarda la hora de término tras buscar la clave de trabajo anteriormente ingresada por el usuario.
- *Usuarios*: son todos aquellos individuos que interactúan con el sistema.
- *Procedimientos administrativos*: conjunto de reglas para los usuarios sobre el manejo correcto del sistema. Por ejemplo: el personal de calidad es el único que accederá a los registros almacenados en la base de datos para hacer algún cambio y/o eliminarlos.

La idea de desarrollar el sistema surge en el momento que los ejecutivos y altos mandos concluyen que la forma actual de almacenar y administrar la información de la empresa se hace insuficiente, el tiempo de respuesta es tardío y no satisface las necesidades que se requieren.

Las etapas de vida de un SI desde su planeación serían, el análisis de requerimientos, la arquitectura y diseño, desarrollo e implementación, pruebas y mantenimiento; sin olvidar que, entre cada una de las etapas se deben ejecutar controles de calidad para minimizar el riesgo de errores.

### 2.3. Ingeniería de software

Aclaremos la diferencia entre ciencia de la computación e ingeniería de software. La primera engloba métodos y teorías que son parte del software y la computadora, mientras la ingeniería de software se refiere a los problemas prácticos de producir software. Aclarado este punto recordemos que en la década de 1980 y posteriores creció en gran manera la influencia y demanda de las aplicaciones referentes a las nuevas tendencias que eran Internet y móviles.

Las grandes compañías querían hacerse siempre notar que sus productos y servicios eran de mayor prestigio y vieron en la computación una puerta para expandir sus límites territoriales. Por ejemplo, con el uso del Internet compartir todo lo que desee una compañía de Estados Unidos a una situada en Japón dejaría de ser imposible, y para lograr este objetivo se requería de un software específico que cumplieran con tareas demandadas por los clientes, creando así el *software hecho a la medida*.

Para evitar grandes crisis a la hora de desarrollarlo, ya que se requiere de una cotización de recursos humanos, materiales, etc. además de un tiempo de desarrollo y de pruebas antes de su lanzamiento, las compañías desarrolladoras de software emplearon metodologías, y las más eficientes se volverían materia de estudio e implementación general en la actualidad.

La ingeniería de software incluye personas encargadas de analizar y desarrollar los estudios necesarios para generar el software todo esto a través de un proceso para cumplir así el proyecto solicitado y entregar el producto al cliente.

Entonces ¿qué es un proceso de software? La respuesta sería un conjunto de actividades y resultados involucrados para producir el producto de software. Dichas actividades se encuentran presentes en todo tipo de proceso involucrado al software, y son:

- Especificación del software: se explican las características que deben cumplirse en el producto y sus limitaciones, todo esto para que al final se cumpla con un tiempo límite y un costo establecido estando de acuerdo ambas partes (contratista y desarrollador).
- Desarrollo del software: plataforma, programación y recursos partícipes en la construcción del producto.
- Evolución del software: se concluyó con el paso de los años que un producto que es capaz de modificarse y/o adaptarse a los requerimientos del cliente y del mercado serían lo más rentables y convenientes en el negocio del desarrollo de software.

Pero no sólo se compone de software un sistema, la ingeniería se encarga de analizar los aspectos de desarrollo y evolución de sistemas socio-técnicos. También involucra el hardware y la interacción del sistema con los usuarios, piensa en los servicios, restricciones y la forma que tiene que funcionar el sistema para cumplir su propósito. Sus objetivos principales son:

- Ver al sistema como un todo: posee una visión global y comprensión de cómo integrar los componentes de forma segura para satisfacer su propósito.
- Considerar el impacto universal: se requiere contemplar el ciclo de vida del sistema y cada fase (diseño, desarrollo, construcción, distribución, mantenimiento y desmantelamiento). Es necesario considerar las decisiones y sus consecuencias de cómo afectará al sistema antes, durante y después de su integración a la organización.
- Esfuerzo multidisciplinario: para el desarrollo de un sistema desde su análisis se involucra un equipo capaz que asegure el cumplimiento de cada uno de los objetivos por los cuales se está creando dicho sistema, guiándose ante todo a un plan de trabajo para evitar reprocesos.

### 2.3.1. Modelos del proceso del software

Existen diferentes procesos del software útiles para el desarrollo de productos que cumplan con las características solicitadas por el cliente para satisfacer su necesidad, tiene que haber un orden diferente o algunas fases más cuidadas que otras a la hora del desarrollo. Así dieron nacimiento a diferentes modelos de procesos de software y de éstos parten variaciones, todo esto para obtener de forma eficaz el resultado. Algunos de estos modelos son:

#### - **Modelo V**

Se distingue por dos partes principales: fases de construcción y fases de pruebas. El desarrollo del sistema comienza con la captura y definición de los requisitos. Los requisitos son capturados del cliente y de los futuros usuarios. Ellos son usados en la siguiente fase para desarrollar los modelos funcionales del sistema. Los modelos funcionales deben ser independientes de la implementación del sistema, para evitar decisiones tempranas de diseño. Esta fase define la estructura del sistema en componentes, y a la vez, las interfaces correspondientes. Seguidamente, el comportamiento detallado de los componentes es definido en la fase del *diseño detallado*. Finalmente, las fases de construcción del modelo V terminan con la implementación de los componentes.

Después en las pruebas unitarias y la implementación de los componentes, son probados a partir de sus especificaciones. En el siguiente nivel, las pruebas de nivel de integración son usadas para probar los módulos terminados, la fase termina cuando todos los componentes responden exitosamente. Y las pruebas de sistema se realizan cuando el sistema está disponible por completo.

#### - **Modelo en Espiral**

Las actividades de este modelo se asemejan a un espiral en la que cada bucle o iteración representa un conjunto de actividades. Si el cliente quiere seguir haciendo

mejoras en el software, se vuelve a evaluar las nuevas alternativas y riesgos y se realiza otra vuelta de la espiral, así hasta que llegue un momento en el que el producto de software desarrollado sea aceptado y no requiere seguir mejorándose con otro nuevo ciclo.

#### - **Desarrollo Evolutivo**

Este proceso de desarrollo entrelaza las fases de especificación, desarrollo y validación. La primera fase es abstracta, porque a medida que se va desarrollando el cliente irá comprobando si es lo que realmente necesita y desea que contenga su producto. El detalle de este proceso es que el tiempo y costo del desarrollo puede crecer si no se tiene cuidado de lo que requiere el cliente y que exista una correcta comunicación con el equipo de desarrollo.

Éstos son algunos ejemplos de los diferentes procesos de modelos de software, cada uno se ajusta al tipo de producto solicitado. Para la aplicación desarrollada en FoxPro, se empleará la metodología en cascada modificada.

## 2.4. Metodología en cascada modificada

La metodología considera cinco actividades fundamentales para el desarrollo de software. Cada una depende de su predecesora, por lo que es un proceso muy minucioso a la hora de emplearlo, no pueden ocurrir modificaciones drásticas como el modelo evolutivo, por ejemplo. Las fases de la metodología en cascada modificada se muestran en la figura 2.1.

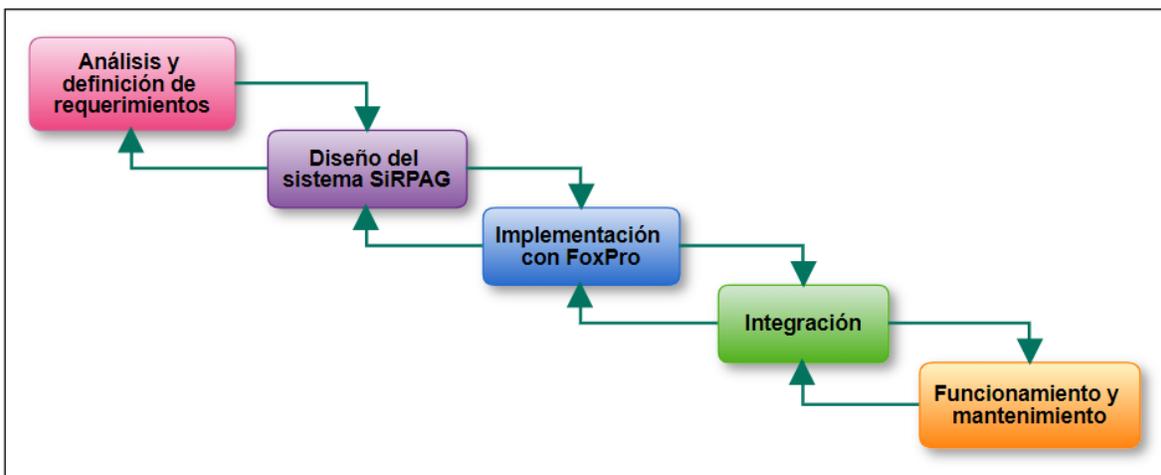


Figura 2.1 Metodología en cascada modificada para el desarrollo de SIRPAG.

- *Análisis y definición de requerimientos*: las características, servicios y metas definidas a partir de lo que el usuario o cliente requiere, se definen los detalles y límites, así como tiempo y recursos necesarios.

- *Diseño del sistema SiRPAG*: este diseño se divide a su vez en hardware y software. Se establece desde aquí la arquitectura completa del sistema y componentes físicos y lógicos describiendo las abstracciones fundamentales del sistema. Los productos principales en la fase del diseño son la estructura de los datos, la arquitectura del sistema, detalle procedimental y por último las interfaces.
- *Implementación con FoxPro*: cada parte que compone al producto final se conoce como unidad y en conjunto generan los componentes del programa que se desarrollará.
- *Integración*: después de probar las unidades que conformarán el sistema, se adjuntan para conformar el sistema completo.
- *Funcionamiento y mantenimiento*: como su nombre lo dice, ver que el sistema cumple con los requerimientos del usuario en el entorno para el que fue destinado crearse. Y el mantenimiento se compone de cuatro tipos que son mantenimiento perfecto, adaptativo, preventivo y correctivo.

Los siguientes temas describirán de manera detallada los componentes de la metodología en cascada modificada.

#### 2.4.1. Análisis y definición de los requerimientos

El análisis de los requerimientos para el desarrollo de un sistema o aplicación es la parte medular en su construcción, al concluir el desarrollo gran parte de los defectos encontrados se pudieron haber evitado desde la toma de los requerimientos. En el análisis hacemos un refinamiento, modelado y especificación del ámbito del problema hasta establecer la estructura del sistema.

Creando así el “Plan del software” que contiene modelos de flujo de información, procesos y contenido de datos. El análisis de los requerimientos se puede identificar con cinco actividades:

- *Reconocimiento del problema del software*
- *Evaluación del problema y síntesis de la solución*
- *Modelado*
- *Especificación*
- *Revisión de especificación*

En los requerimientos el objetivo principal es expresar *qué* se supone debe hacer la aplicación, el *cómo* vendrá a partir de estos lineamientos. La salida de este análisis es un documento conocido como *Especificación de Requerimientos de Software (ERS)*.

El análisis de los requerimientos se divide en dos tipos: los requerimientos del cliente, como su nombre lo dice expresa las necesidades del cliente en un lenguaje claro para él; el segundo, consiste en los requerimientos del desarrollador donde se dan las especificaciones y estructura más detallada de lo que se tiene que hacer.

La importancia de documentar los requerimientos es para definir las metas que se tienen que cumplir, límites de la aplicación, tiempo en los procesos (diseño, programación, implementación, pruebas, integración e instalación) y el tamaño del esfuerzo en cada uno de los módulos que se desarrollarán.

**El mapa conceptual del análisis de requerimientos** es el conjunto de las necesidades del cliente más las del desarrollador documentadas, integrando en ellas el tiempo que se dedicará a cada requerimiento y la cantidad de trabajo que tomará.

Esta documentación debe ser tratada con mucho cuidado, si desde este punto no se localizan los defectos ni inconsistencias, terminado el producto los errores de diseño son costosos y lamentables para el proyecto ya que se pudieron haber prevenido desde el comienzo del diseño. Las pruebas son un paso que depende también de los requerimientos, así sabremos si el producto está cumpliendo con los requerimientos del cliente. Los puntos primordiales del mapa conceptual son los siguientes:

### **1. Identificar al cliente**

Siguiendo el primer punto para hacer el mapa conceptual de los requerimientos identificaremos al “dueño de la compañía” y también a los *Interesados* que tienen el deseo de conocer los resultados del producto, todos componen al cliente que, por una o varias razones, desean que se lleve a cabo la creación del producto.

El cliente tiene una lista de “deseos” que quiere que cumpla el producto, por lo que es fácil que entren en conflicto algunos requerimientos que pudieran quedar inconsistentes. Por consiguiente, el líder del proyecto tiene que llegar a una *negociación*, el producto final tiene que ser alcanzable y con un tiempo finito para desarrollarse, sin exceder los costos estimados por parte del cliente o podría quedar el proyecto inconcluso o cancelado por completo.

### **2. Entrevistar al cliente y sus representantes**

Es común que el cliente todavía se encuentre formando ideas de lo que quiere y necesita, por lo que a la hora de tomar esos conceptos es necesario plantearlos en una lista y trabajar con el cliente para aclarar si son todos los deseos y necesidades que requiere.

Las necesidades son más fáciles de clasificar que los deseos del cliente, debido a que se pueden incluir como parte obligatoria en el desarrollo del producto, por ejemplo, que exista un botón en todo momento para poder salir de la aplicación sin que guarde los cambios por si se llegara a cometer algún error, esto más que un deseo es algo que necesita llevar la aplicación y que daremos como requerimiento para el desarrollador y a su vez identificaremos que hardware se necesitará.

### **3. Escribir los requerimientos del cliente en forma de documento estándar**

Aunque la solicitud para obtener un sistema viene del cliente, el entrevistador (el administrador del proyecto) irá formando una visión conjunta con el entrevistado del nivel de alcance del sistema.

Inicialmente en la entrevista se realizan notas exhaustivas y un bosquejo de los requerimientos obtenidos a través de un estándar, dependiendo de lo que requeriremos para plasmar y abrir el panorama de lo que se piensa desarrollar es que utilizaremos casos de uso, flujo de datos y/o diagramas de estado si así lo requiere el tipo de producto solicitado. Al final se realiza una firma de acuerdo para comprometer a ambas partes al cumplir las condiciones descritas en el documento.

#### 4. Inspeccionar los requerimientos dados por el cliente para su aprobación

Los diagramas ayudan a plasmar y dar una visión consistente de lo que pide el cliente y será importante para finalizar la toma de requerimientos del mismo (recordemos que también se tienen que hacer los requerimientos del desarrollador). Al término de la documentación se envía un correo informativo a los involucrados recibiendo sus comentarios que retroalimenten la documentación hasta obtener su validación.

#### 5. Construir los requerimientos del desarrollador

Obtenida la aprobación del cliente con respecto a los requerimientos procedemos con la toma de requerimientos con el(los) desarrollador(es) que contendrá un lenguaje más técnico comparado con los elementos recabados con el cliente, para comprender lo que necesita el producto y enriquecer la documentación para disminuir las probabilidades de un mal entendido.

### 2.4.2. Diseño

El diseño son aquellas interconexiones que tendrán las diferentes unidades desarrolladas del sistema, *es parte de la arquitectura pero detallando los componentes, mientras que la arquitectura muestra la estructura general.*

La arquitectura se obtiene mediante un proceso de partición del análisis de los requerimientos a soluciones en software como podemos ver en la figura 2.2.

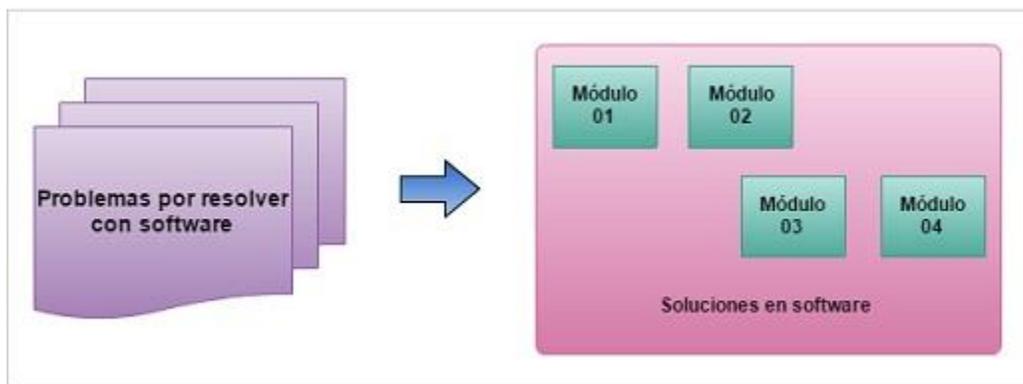


Figura 2.2 Descomposición de los requerimientos.

Tras la toma de los requerimientos si decidiéramos omitir el paso del diseño del software estaríamos arriesgándonos a construir un sistema inestable, incompleto y obsoleto, al producir

cambios no seríamos capaces de visualizar el impacto que haría al software y después todas esas implicaciones incrementarían costos en reestructuración del código.

El diseño de software se traduce a tres etapas que producen cuatro productos principales para el desarrollo correcto del software y son:

- **Diseño preliminar** centrado en convertir los requerimientos en la **arquitectura del software**. Nos podemos respaldar con diagramas de caso de uso y diagrama de actividades para describir la estructura general del sistema.
- **Diseño detallado** de todas las especificaciones de los datos y algoritmos. En esta etapa se desarrollan y documentan la *estructura de los datos*, *estructura del programa* y *los procesos* con el fin de continuar con los pasos siguientes: codificación y pruebas.
  - *Diseño de los datos*: transforma los requerimientos obtenidos en la estructura de datos que se va a implementar en el sistema. Se hace el uso de diagramas de flujo de datos, modelado de máquinas de estado, diagrama entidad-relación y un diccionario de datos para la descripción de la base de datos.
  - *Diseño arquitectónico*: establece las relaciones de control entre los módulos del sistema y que facilite el flujo de información. Se usa la jerarquía de control para plasmar la dependencia de los módulos entre sí.
  - *Detalle procedimental*: después del diseño de los datos y la estructura sigue la descripción de los procesos del sistema. Todo proceso tiene una secuencia (procedimientos), condición y repetición. A partir de la jerarquía de control se extrae el procedimiento del software descrito con diagramas de flujo y diagrama de jerarquía de clases.
- **Diseño de la interfaz** para determinar la comunicación entre sí de los diferentes componentes del sistema y del usuario con el sistema. Para la primera se hace el uso del diagrama de interfaz de navegación y el uso de las reglas heurísticas de usabilidad y el segundo con ventanas de interfaz de usuario.

Otro factor a tomar en cuenta en el diseño es la calidad de mismo, para esto acudimos a una serie de técnicas formales que tienen como objetivo descubrir errores en la lógica e implementación de cualquier abstracción, cumpliendo con los requerimientos y garantizar que se está satisfaciendo los puntos del ciclo de vida establecido.

#### 2.4.2.1. Bases de datos

Las bases de datos son un conjunto de datos relacionados, diseñada para satisfacer las necesidades de información de una organización. Desde la toma de requerimientos se hace mención del almacenamiento de los datos, pero en el diseño se determina su estructura con el uso del diagrama entidad-relación.

**Las operaciones** que podemos realizar sobre las bases de datos son de tipo DDL<sup>1</sup> y DML<sup>2</sup>. El primero como su nombre lo dice, describe y da nombre a los datos que se requieren junto a las reglas de integridad y seguridad. El segundo está enfocado a consultar, añadir, modificar o borrar datos de la base de datos.

---

<sup>1</sup>Data Definition Language (Lenguaje de Definición de Datos)

<sup>2</sup>Data Manipulation Language (Lenguaje de Manipulación de Datos)

Los Modelos de datos determinan esta estructura lógica y entre éstos están: el modelo entidad-relación y el modelo relacional.

- *Modelo entidad-relación*: que representa los datos tal y como se captan en el mundo real.
- *Modelo relacional*: además de la descripción de entidades y sus atributos una restricción importante es las entidades con las que otra entidad se puede asociar a través de *relaciones*.
  - **Relación de uno a varios (1,n)**: presente cuando cada registro de una tabla está enlazado con varios registros de una segunda tabla, pero cada registro de la segunda tabla está enlazada con un único registro de la primera tabla.
  - **Relación de uno a uno (1,1)**: en este, un registro de la primera tabla se encuentra sólo relacionado con un registro de la segunda tabla y viceversa.
  - **Relación de varios a varios (n,m)**: cada registro de la primera tabla esta enlazado con uno o varios registros de la segunda tabla y viceversa. Para evitar esta situación se necesita crear una tabla intermedia que esté relacionada con las dos tablas.

El lenguaje frecuente para generar las bases de datos relaciones es SQL (Structured Query Language) estándar implementado por la mayoría de los sistemas de gestión de bases de datos relacionales.

Además de ser un lenguaje de consultas, incluye características para definir la estructura, modificación y restricciones de seguridad de los datos. Se resumen en tres cláusulas la estructura básica de SQL:

- **SELECT**: usado para listar los atributos deseados para una consulta.
- **FROM**: define las tablas de las que se realizarán las instrucciones.
- **WHERE**: se usa para determinar los registros de la lista de tablas de la cláusula *FROM* que aparecerán en los resultados de la cláusula *SELECT*.

Lenguaje de definición de datos:

- CREATE
- ALTER
- DROP
- TRUNCATE

Lenguaje de manipulación de datos:

- ORDER BY
- INSERT
- UPDATE
- DELETE

Al término de los modelos de entidad-relación y modelos relacionales se requiere aplicar una serie de reglamentos, a esto se le conoce como **normalización**. Enfocado a comprobar la validez de los esquemas lógicos, no obtener datos redundantes y tener máxima estabilidad. Según (Silberschatz, 2006, p. 219) existen varias formas de normalización, pero las que se presentan en la mayoría de los procesos de perfeccionamiento en las bases de datos son las siguientes:

- **1.<sup>a</sup> Forma Normal**

La primera forma impone que los datos contenidos en nuestras columnas de la tabla sean atómicos, lo que significa que dentro de una tupla no contendrá más de un objeto único. Por ejemplo, la columna de dirección de un usuario que contenga información como la calle, número, colonia, entre otros aspectos, no será considerada como objeto único, la primera forma indica que los elementos del registro son unidades indivisibles.

Otro de los elementos que señala la primera forma es que los registros de una columna sean del mismo campo semántico, las columnas son de un nombre único y no debe haber datos repetidos entre dos o más columnas.

- **2.<sup>a</sup> Forma Normal**

La relación de dos tablas cumplirá con la segunda forma si satisface la primera y además todo atributo no-llave depende únicamente de su llave primaria.

- **3.<sup>a</sup> Forma Normal**

Después de haber cumplido las dos primeras formas normales, la relación de columnas en la tabla no debe ser transitiva. Por ejemplo, si tenemos la columna fecha de nacimiento, edad y una que señale si es mayor de edad, ésta última estaría presentando un caso de columna transitiva y tendría que ser eliminada.

- **Forma Normal Boyce-Codd**

Ésta forma es una versión ligeramente más fuerte que la tercera. Tiene que cumplir con las tres anteriores y además tener claves candidatas compuestas sin contar con atributos en común.

Por ejemplo, tener una columna con el id de los doctores pertenecientes a un hospital, su curp e id de la secretaria que tienen asignada, si sólo se les asignara una secretaria por doctor cumplirían con la forma. Pero, si la secretaria puede tener hasta dos doctores asignados, entonces la clave candidata de id del doctor tiene un dato en común con otro registro de la misma columna y ya no cumpliría con la forma.

#### 2.4.2.2. Interfaz

Para comunicarse entre sí los componentes (unidades) que conforman al sistema y acoplarlos (que cuenten con la relación y comunicación) es requerida una interfaz. Ésta se encuentra delimitada por la comunicación que requieran sus componentes, no debe contener accesos innecesarios, ser independientes y contener una cohesión funcional y secuencial.

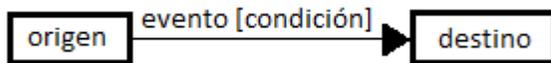
Los clientes comprenden una aplicación al visualizar su interfaz gráfica por lo que primeramente hay que diseñar el **Diagrama de Navegación de Interfaz (DNI)** y describir todas las **ventanas de Interfaz de Usuario** (ventanas IU), terminado esto es necesario aplicar **las reglas heurísticas** de usabilidad de Jakob Nielsen para así evaluar la calidad del diseño de la interfaz de usuario y reestructurarla con el fin de obtener una interfaz adecuada.

- Las ventanas IU consisten en describir a través de una representación gráfica llamada “Ventana IU” cada pantalla que se visualiza al ejecutarse la aplicación, en su diseño se aplican los mismos íconos empleados en una interfaz gráfica de usuario (botones, textfield, listas desplegables, etc.), e identificar estos componentes, sus atributos y acciones que producen, cuando se utilizan y describirlas en una tabla IU.
- El DNI describe el orden y tipo de pantallas que se van produciendo durante la ejecución de la aplicación. A través de un grafo dirigido en el que cada nodo sea una Ventana IU y cada flecha que une dos nodos es la acción que produce la transición de una ventana a otra. Sus elementos son los siguientes:

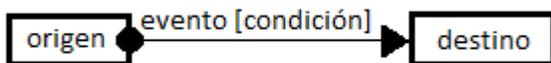


Y sus transiciones:

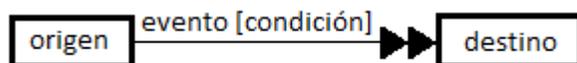
- Normal: cuando el evento origina la apertura de una ventana sin restricción.
- Eventos paralelos: esto ocurre cuando el evento provoque la aparición de más de una ventana.
- Transiciones opcionales: dado un evento, la transición de una ventana a otra dependa de una condición (señalada entre corchetes a continuación del evento).



- Cierre de ventanas: cuando ocurre la transición de una ventana a otra, la ventana origen se cierra dejando sólo la ventana destino.



- Ventanas modales: la ventana destino puede aparecer de dos formas. La primera puede ocurrir que sea modal (bloqueante) y mientras está presente será la única que podrá usar el usuario; la segunda es la *no modal* donde permite utilizar todas las ventanas abiertas al usuario. Cuando es modal la transición se describe con una flecha doble. Es necesario destacar que no puede haber más de una ventana modal en las transiciones paralelas debido a su propiedad bloqueante.



A partir de las Ventanas IU como sus DNI aplicaremos **las reglas heurísticas de usabilidad de Jakob Nielsen** para detectar los problemas posibles de usabilidad de la aplicación y facilitar su uso a los usuarios. Las reglas son las siguientes:

1. *Visibilidad del estado del sistema.* El sistema debe mantener informado al usuario del estado del mismo, con la retroalimentación apropiada en un tiempo razonable.
2. *Utilizar el lenguaje de los usuarios.* El uso de palabras, frases y conceptos familiares en lugar de términos orientados al sistema de tal forma que la información aparezca en un orden natural y lógico.
3. *Control y libertad para el usuario.* En el dado caso que el usuario cometa un error o desee revertir algún cambio realizado necesitará una salida de emergencia para salir de ese estado indeseado evitando mayores complicaciones con comandos como: cancelar, salir del sistema, edición, etc.
4. *Consistencia y estándares.* Usar los mismos términos y no cambiarlos a lo largo de la interfaz ya que pudiera confundir al usuario haciéndolo pensar que efectúan diferente acción. Por ejemplo: primero usar “Salvar” a la hora de guardar la información y después “Salvar fichero” o “Salvar registro”. También situar los controles en el mismo sitio en cada ventana.
5. *Prevención de errores.* El sistema sería deficiente si en pocos pasos ya estuviera arrojando mensajes de error.
6. *Reconocer mejor que memorizar.* Es mejor mantener visibles los objetos por seleccionar que buscarlos después de memorizar que hacer, si algunos no cambian mantenerlos como valores por defecto.
7. *Flexibilidad y eficiencia de uso.* Las instrucciones para el uso del sistema deben ser fáciles de acceder cada vez que se necesiten, esto para ayudar al usuario inexperto pero que no sea un estorbo para el experimentado.
8. *Los diálogos estéticos y diseño minimalista.* No debe contener información que sea inaplicable o necesaria raramente. Tiene que ser concisa y entendible desde los primeros segundos de ser vista por el usuario.
9. *Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores.* Los mensajes de error deben ser simples, entendibles y que indiquen exactamente el problema para poder así retroalimentar al usuario.
10. *Ayuda y documentación.* Cuando sea necesario la ayuda ésta sea fácil de localizar, entendible y no muy extensa.

### 2.4.3 Implementación con Visual FoxPro

Visual FoxPro es un lenguaje de Programación Orientado a Objetos (POO) que proporciona herramientas para crear y gestionar aplicaciones de bases de datos de 32 bits. Contiene un DBMS: Database Management System<sup>3</sup> y un Entorno de Desarrollo Incorporado fabricado por Microsoft, Visual FoxPro 9.0 es la última versión publicada en el 2007 y que posee integración a Visual Studio .NET y SQL Server. Permitiendo excelente compatibilidad ascendente y logrando así un mantenimiento evolutivo a las aplicaciones realizadas en FoxPro.

Con un conjunto de herramientas para la construcción de soluciones de bases de datos para escritorio, entorno cliente-servidor, o en la Web, capaz de gestionar información de la organización, ejecutar consultas y crear un sistema de gestión de bases de datos relacionales a través de una programación adaptada a las necesidades de los usuarios.

Componentes de interés:

*Acceso universal a los datos y procesamiento de datos:* con la clase CursorAdapter permite acceder a datos nativos de Visual FoxPro, XML, SQL Server, o cualquier otra fuente mediante ODBC u OLE DB.

*La gestión estructurada de errores:* mecanismo de herencia de excepciones, el control y encapsulado de errores en el código mediante el uso de estructuras TRY CATCH y mucho más.

*Gestor de eventos:* permite crear y modificar eventos y métodos que desencadenan una serie de instrucciones como respuesta a alguna acción del usuario.

*Informes del proceso y ejecución de la programación:* nueva arquitectura de informes de errores, proceso, gestión de tablas y datos. Con múltiples bandas de detalle sobre cómo se van procesando y ejecutando las diferentes operaciones y métodos con vista preliminar.

Visual FoxPro no consume demasiados recursos, los requerimientos mínimos son:

- PC con procesador Pentium o superior
- Sistema Operativo Microsoft Windows 2000 o superior (adaptado por parte de sus usuarios para ser compatible con Windows 10)
- Microsoft Windows Server 2003 o superior
- Memoria RAM de 64 MB mínimo
- Disco duro de 165 MB para su instalación y 20 MB adicionales para los requisitos de Microsoft Visual FoxPro 9.0
- VGA 800 x 600 px con 256 colores o superior
- Mouse

Con este mínimo de elementos se puede montar un sistema que cumpla con las necesidades de Excelform, siendo una empresa mediana no es alta la demanda de recursos y equipos de cómputo para cada área por lo que esta tecnología no puede ser vista como una limitante a corto y mediano plazo, posteriormente al tener integración con Visual Estudio se podrá migrar la aplicación a una plataforma contemporánea. Con el uso de FoxPro se estará cumpliendo con uno de los requerimientos solicitados por el cliente.

---

<sup>3</sup>Sistema Gestor de Bases de Datos

#### 2.4.4. Integración

Esto ocurre al incorporar nuestro sistema desarrollado en el entorno que corresponde, cuando se crea (fase de implementación) se trabaja en un ambiente controlado y conocido por los desarrolladores, al concluir dicha fase tenemos que realizar su correspondiente instalación.

La instalación del producto terminado es un proceso que consta de llevar el sistema al equipo de cómputo destino o entorno de trabajo para el que fue solicitado, configurándolo para ser empleado por el usuario final.

Para la correcta instalación del producto se dependerá de varios factores, desde el desarrollo, las dimensiones del programa, su accesibilidad y nivel de distribución: descarga electrónica, a través de Internet, disco duro, memoria flash, etc. Las normas de instalación son necesarias para limitar el número de individuos que tendrán derecho a usar el sistema, conociendo los detalles del producto para mejor desempeño y evitar problemas de incompatibilidad de hardware y software. Los puntos esenciales que se tienen que considerar a la hora de instalar el software son:

- Criterios del equipo como especificaciones del procesador, memoria RAM, disco duro, sistema operativo, etc.
- Actualización del sistema, drivers, o fuentes que se requieran para completar la instalación.
- Configuración requerida para comunicar al equipo con el servidor (modo multiusuario).

La **validación** se refiere a si se satisfacen los requerimientos según se establecen en las especificaciones del cliente. La validación pregunta si se “construye lo correcto”. Todo esto se presenta desde que se van desarrollando las unidades del sistema, todo tienen que estar validado y demostrado que es funcional, óptimo y que cumple con lo solicitado por el usuario.

No confundamos validación con *verificación*. Este último se refiere a si se está construyendo bien aquellas unidades que se especificaron en una etapa anterior. La verificación pregunta si se está “construyendo bien”.

#### 2.4.5. Funcionamiento y mantenimiento

Este proceso tiene el mismo nivel de importancia que el resto del ciclo de desarrollo de un sistema. El objetivo es la modificación del producto, pero manteniendo su objetivo de desarrollo e integridad todo esto para mantener actualizado y en correcto funcionamiento, siendo el final de este proceso cuando el producto es completamente retirado de donde se realizó la instalación.

El mantenimiento del hardware implica el remplazo de un componente o de toda una estructura compatible con el entorno, para optimizar su funcionalidad a partir de las necesidades que el mismo usuario requiera satisfacer. En el ámbito del software este se clasifica en diferentes tipos de mantenimiento:

- **Mantenimiento preventivo:** se enfoca en las mejoras que se realizarán en el producto sin alterar sus especificaciones funcionales, se incluye una serie de sentencias que permitan validar los datos de entrada y salida para comprobar que sus resultados sean los correctos.
- **Mantenimiento perfectivo:** conjunto de actividades que mejoren o añadan nuevas funciones que requiera el usuario. En el software se definen dos tipos de mantenimiento perfectivo.
  - a. *Mantenimiento de ampliación:* adición de componentes que poseen nuevas funcionalidades.
  - b. *Mantenimiento de eficiencia:* se realizan mejoras de componentes ya existentes para optimizar su funcionalidad.
- **Mantenimiento adaptativo:** con cada cambio que ocurra en el entorno de ejecución ya sea de hardware o software, esto implicara de un simple retoque a toda una reescritura del código para adaptarlo a su nuevo medio.

Éste es el tipo de mantenimiento más recurrente de todos y se debe a la actualización constante de sistemas y herramientas en el software a corto plazo. Por lo que a su vez, el mantenimiento adaptativo se divide en dos tipos: mantenimiento en el entorno de los datos (ej. cambio de tablas libres a una base de datos relacional) y mantenimiento en el entorno de los procesos (ej. traslado a una versión actualizada de la plataforma de desarrollo).

Además, mientras más antiguo sea el software más costoso será el realizar su mantenimiento como se puede ver en el ejemplo de la figura 2.5, conforme pasan los años la organización comienza a crecer por lo que se requiere migrar, o en el común de los casos comprar un nuevo software a la medida que satisfaga las necesidades actuales de los usuarios.

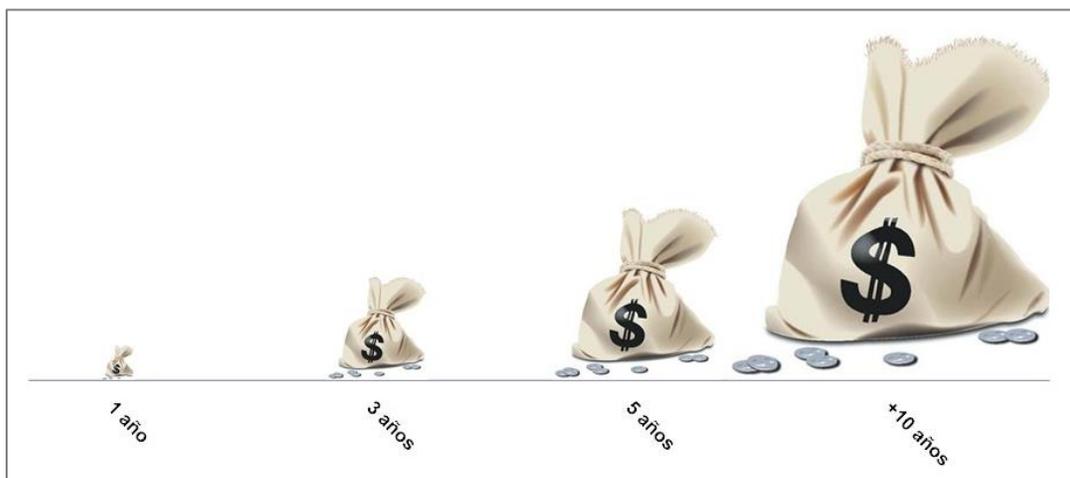


Figura 2.3 Evolución del costo del mantenimiento de un software partiendo desde su instalación.

- **Mantenimiento correctivo:** cuando el comportamiento del sistema no es el esperado quiere decir que se está presentando un fallo, el objetivo de este tipo de mantenimiento es para que no presente más y a su vez eliminar cualquier otro tipo de errores en el software. Los fallos pueden ser de distintos tipos: desde el fallo en el procesamiento de los datos de entrada, lenta respuesta del sistema, incongruencia del software contra el manual de usuario, etc.

#### 2.4.5.1. Dificultades en el mantenimiento

El elemento primordial que se debe mantener intacto a la hora de efectuar algún tipo de mantenimiento es la calidad del software. Debe seguir cumpliendo con todas las especificaciones por las cuales se creó, por lo que el código heredado es uno de los problemas a los que se enfrentan los programadores a la hora de hacer los cambios necesarios.

El código heredado hace referencia a todo código dentro del programa que se desarrolló tiempo atrás y por personas que actualmente no se encuentran en el equipo de mantenimiento, siendo probable que la documentación de desarrollo y mejoras no se encuentre actualizado y complique demasiado las cosas a la hora de realizar los cambios requeridos.

Los puntos indispensables a la hora de hacer mantenimiento al software son:

- **Continuidad del cambio.** Es natural que para seguir usando un programa éste reciba cambios y mejoras para ser eficiente. Esto nos lleva al punto en el que la complejidad del desarrollo crece y es necesario utilizar técnicas de ingeniería del software para hacerlas llevaderas y entendidas al equipo responsable en ese momento.
- **Evolución del programa.** A pesar de su evolución se tienen que contemplar los siguientes puntos:
  - *Documentación desfasada:* a la hora de hacer el mantenimiento el estilo del programador a veces no es comprendida por el resto de los integrantes, por lo que se requiere una documentación con los cambios introducidos.
  - *Programa menos estructurado con cada cambio:* se presentan efectos secundarios en el diseño, rendimiento, operaciones lógicas, redefinición o adición de variables locales o globales, cambio de formato sobre los datos o archivos y hasta la documentación se ve involucrada, por lo que es necesario recurrir a estándares y técnicas de programación para reducir este impacto con cada cambio que se vaya sumando.
- **Conservación de la estabilidad en el desarrollo.** No conviene y no es una opción el que un cambio tome más tiempo y costo comparándolo al desarrollo de un módulo de la misma complejidad elaborado en la fase de implementación del sistema.

Por último, es importante conservar la **mantenibilidad** del programa tras los cambios. Esto se define como la facilidad de comprender, planificar, corregir y adaptar el software, se requiere tener cuidado y atención a cada uno de los puntos vistos párrafos atrás para en lo subsecuente se pueda facilitar el mantenimiento hasta que se termine el ciclo de vida del software.

## 2.4.6. Pruebas

Las pruebas tienen como objetivo el cerciorarse que el sistema hace lo que el usuario le solicita de la forma correcta. Un plan de prueba ayudará a diseñar y organizar los test con el fin de evaluar que los módulos se ejecuten de forma correcta. Se deben plantear los siguientes puntos:

- Determinación de los objetivos de prueba
- Diseño de los casos de prueba
- Preparación escrita de los casos de prueba
- Verificación de los casos de prueba
- Ejecución de las pruebas
- Evaluación de los resultados

Las pruebas de unidad verifican que cada unidad que conforma al sistema sea correcta y funcional, este proceso se puede llevar a cabo en paralelo para múltiples módulos. Se prueban los siguientes aspectos:

- *Interfaz del módulo*: para verificar si la información fluye de forma adecuada dentro del módulo que este siendo evaluado. Revisando entrada, validación y almacenamiento de datos.
- *Estructura de datos*: son para cerciorar que los datos que se mantienen temporalmente conserven su integridad durante la ejecución del programa.
- Las *condiciones límites*: así determinar qué tiene como restricciones de procesamiento y resultado.
- *Bifurcaciones o caminos independientes*: todos y cada uno de los módulos que componen el sistema son partes esenciales y que se ejecutan por lo menos una vez.
- El *manejo de errores*: para asegurar de que el sistema no se detenga bruscamente y preferentemente concluya la ejecución enviando errores controlados.

Tenemos diferentes tipos de pruebas, y en cada una se entrega una descripción de los resultados esperados y obtenidos.

*Prueba de camino básico*. Es un método para ejecutar por lo menos una vez cada unidad del programa.

*Prueba de integración*. Una vez que todas las unidades han sido probadas por separados, ahora hay que ponerlas juntas, por lo que la interacción de unidades puede ocasionar que el sistema no funcione bien ya como un todo.

*Prueba de validación*. Son pruebas que tratan de comprobar la conformidad de los requerimientos, la portabilidad, compatibilidad, recuperación de errores, documentación, facilidad de mantenimiento, etc.

*Pruebas del sistema.* Son realizadas para ver qué sucede en situaciones externas al sistema, por ejemplo, el hardware, el tipo de información, las fallas de energía eléctrica que puedan surgir, la seguridad de acceso al mismo sistema, la resistencia a situaciones anómalas, al rendimiento del software, etc.

*Prueba de seguridad.* Son mecanismos de protección incorporados al sistema, lo protegen de penetraciones impropias. Se trata de hacer invulnerable al sistema de cualquier ataque frontal, por los flancos o retaguardia. Los ataques tratan de conocer la información, alterarla o modificarla, dañarla, etc., pudiendo ocasionar fallas al software como al hardware.

*Prueba de resistencia.* La prueba ejecuta un sistema de forma que demande recursos de cantidad, frecuencia o volúmenes anormales, recursos como: interrupciones, frecuencias de entradas, memoria RAM, memoria virtual, excesivas búsquedas de datos, etc. Se intenta tirar el sistema.

*Prueba de rendimiento.* Las pruebas están diseñadas para probar el rendimiento del software en tiempo de ejecución dentro del contexto de un sistema integrado.

*Prueba caja blanca.* Son pruebas que utilizan las estructuras de control para derivar los casos de prueba, son pruebas a nivel componente.

*Prueba caja negra.* Estas pruebas se realizan a nivel entorno, por ejemplo: verificar el correcto funcionamiento de las clases, valores de entrada, tasas de datos y el volumen de tolerancia del sistema, pruebas de comportamiento y el desempeño del sistema, etc.

Los SI son una herramienta primordial para recopilar y procesar información y pueden proporcionar al usuario final datos relevantes para cumplir con algunas de sus necesidades.

La ingeniería de software incluye el estudio de todos los elementos necesarios para desarrollar un SI aplicando una metodología de desarrollo.

La metodología usada para este proyecto es la cascada modificada, confiable por definir de manera rigurosa los lineamientos para el desarrollo factible del producto.

Cada una de las fases tiene una tarea por cumplir, desde el análisis y definición de requerimientos, diseño, implementación, integración, funcionamiento y mantenimiento, todas y cada una de ellas pasan por una serie de validaciones y pruebas para obtener los resultados deseados.

# **Capítulo 3**

# **Caso**

# **práctico**

### 3.1 Inicio del proyecto: especificación de requerimientos de software

#### 3.1.1 Introducción

##### a. Propósito

Reemplazar el papel de las bitácoras de trabajo del personal de producción en captura digital a través de un sistema de reportes, desarrollado con el lenguaje de programación que utiliza la Empresa de Artes Gráficas. Almacenando los datos en una base de datos y ser consultada por los interesados a fin de tener en tiempo los reportes de estatus de maquinaria y Orden de Trabajo (O.T.).

La especificación de estos requerimientos está orientada al personal de Calidad, gerentes de Producción, jefe de Operaciones y director de la Empresa.

##### b. Alcance del producto

El nombre del producto es Sistema Administrador de la Producción en Empresa de Artes Gráficas, SiRPAG para mayor comodidad. Guardará los datos de tiempos de producción y mostrará tablas de consulta que podrán ser descargadas en cualquier momento, permitiendo al analista de operaciones administrar los tiempos de carga de trabajo de las máquinas y operadores.

El personal que estará involucrado en el desarrollo del proyecto se muestra en la tabla 3.1.

Tabla 3.1 Responsables

Área	Funciones	Información de Contacto
Dirección Director General	Planeación y Autorización	<i>correoelectronico@direccion.com</i> Tel.: XX-XXXX-XXXX
Operaciones Jefe de Operaciones	Planeación y Autorización	<i>correoelectronico@direccion.com</i> Tel.: XX-XXXX-XXXX
Calidad Supervisor de Calidad	Planeación	<i>correoelectronico@direccion.com</i> Tel.: XX-XXXX-XXXX
Producción Gerente de Producción de Formas Continuas	Planeación	<i>correoelectronico@direccion.com</i> Tel.: XX-XXXX-XXXX
Producción Gerente de Producción de Dato Variable	Planeación	<i>correoelectronico@direccion.com</i> Tel.: XX-XXXX-XXXX
Sistemas Analista Programador	Documentación, desarrollo, instalación y capacitación.	<i>correoelectronico@direccion.com</i> Tel.: XX-XXXX-XXXX

### c. Referencias

Descripción de palabras incluidas que necesitan ser conocidas en la tabla 3.2.

Tabla 3.2 Palabras clave en los requerimientos

<b>Título</b>	<b>Descripción</b>	<b>Localización</b>
Servidor SRVPDC	Servidor destinado para el uso compartido de aplicaciones y archivos autorizados de la empresa.	Site de la empresa, administrado por el área de sistemas.
Sistema de cotizaciones Mecano	Da de alta los pedidos y genera su cotización, allí se genera la O.T. a través de una tabla en su base de datos que compartirá información a SiRPAG.	Servidor de administración en el site de la empresa, administrado por el área de sistemas.
Visual FoxPro	Lenguaje de Programación Orientado a Objetos con el cual se desarrollará SiRPAG. Licencia comprada y usada hasta la fecha.	Equipos de cómputo del site de la Empresa.
Formulario	Interfaz digital (o documento físico) donde se introducen datos estructurados para poder ser almacenados y procesados.	Hojas impresas cuyo uso es ser la bitácora escrita de los operadores hoy en día. Serán remplazadas por SiRPAG que se asemeja a un formulario electrónico.
XLS	Formato que corresponde a las hojas de cálculo de Microsoft Excel.	Tipo de archivo usado por supervisores para generar sus reportes de producción.
Apache OpenOffice	Software libre que permite abrir archivos de Microsoft Excel.	En todos los equipos del área de producción.
PDF	Formato de almacenamiento para documentos digitales independiente de plataformas.	Todo equipo de la planta posee software que permite visualizarlo.

### 3.1.2 Requerimientos del cliente

#### a. Perspectiva general de producto

Existe un sistema de cotización para dar de alta pedidos en la empresa Excelform, de allí parten la O.T. para el área de producción pero que pronto concluirá su ciclo de vida. Se está migrando actualmente a un nuevo sistema (llamado Mecano), pero en lo que se concluye el proyecto se sigue utilizando el sistema anterior. Adicional a ello se requiere también que exista un sistema donde se capturen los reportes de producción que se habían estado llenando a mano siempre. La representación gráfica del proceso se puede ver en la figura 3.1.

Para poder dar de alta una nueva bitácora de trabajo se requiere el número de O.T., SiRPAG deberá incluir un apartado para la información O.T. generadas en el sistema de cotización en lo que se concluye la construcción del nuevo sistema que alimentará automáticamente el registro de las O.T. para los reportes de producción en SiRPAG.

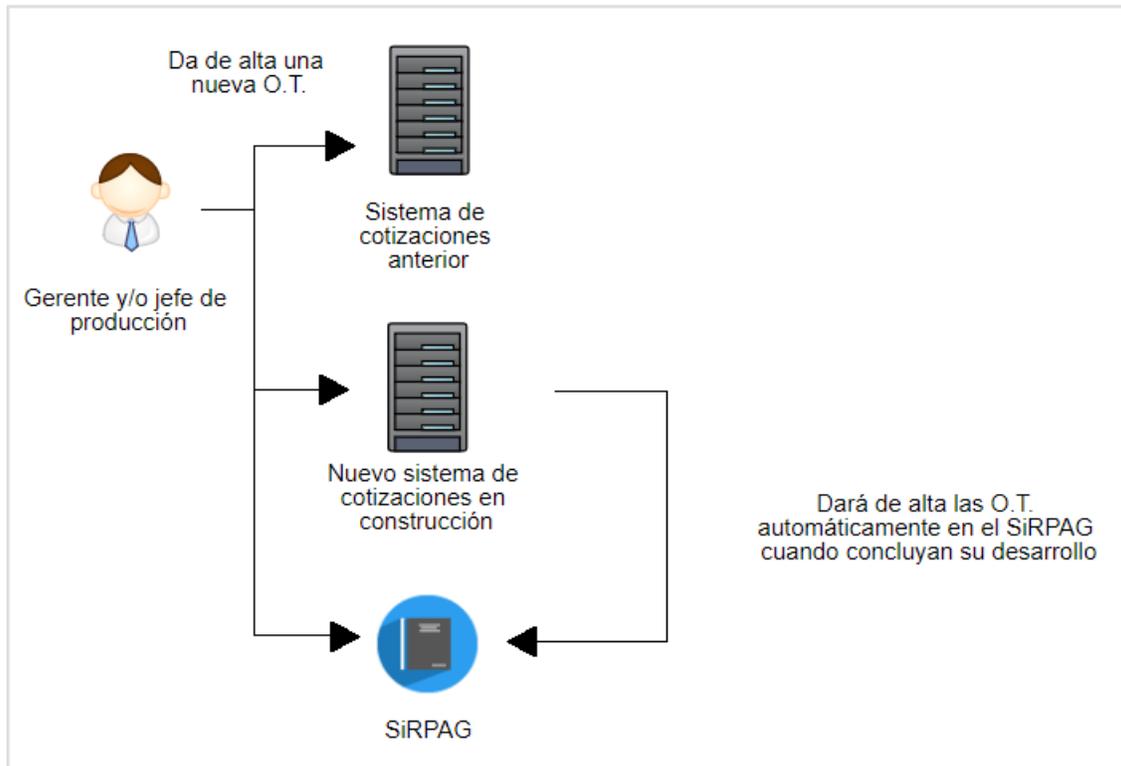


Figura 3.1 Diagrama general de comunicación entre sistemas.

## b. Funcionalidad

El sistema tiene que contar con las siguientes funciones principales:

- Una ventana que solicite al usuario su nombre y contraseña para autenticar que sea un empleado de la empresa e inicie sesión.
- Una opción para poder hacer cambio de su contraseña y consultar su historial.
- La aplicación tiene que mostrar una serie de pestañas fáciles de localizar y que sólo personal autorizado pueda acceder a ellas:
  - Actualizar actividad

- El usuario seleccionará cada una de las actividades que involucren a una O.T., así como la máquina o procedimiento de acabado que se haya empleado.
  - Cada vez que el operador quiera dar de alta una nueva actividad tendrá que cerrar el anterior, excepto para el área de impresión variable y rígidos, ya que los operadores llegan a realizar hasta dos actividades al momento.
  - Lo único que tendrá que ingresar el usuario será la O.T., seleccionará la actividad, la máquina que utilizará (para los de acabado manual seleccionarán el tipo de trabajo en acabado en una lista adicional) y escribirá sus observaciones y cantidad de producción (si es el caso), todo lo demás se almacenará de manera automática: Fecha, Hora de registro (inicio y término), nombre del operador, Área de trabajo, Tiempo total de la actividad.
  - En el caso del área de rígidos se llenan otros campos: largo y ancho del material (se calcula el área), tipo de material, número de pruebas y total de tinta cargada si la actividad lo requiere.
  - El tiempo total de la actividad se calculará automáticamente al obtener la hora de término.
  - Si algún usuario olvido registrar el término de alguna actividad, cuando los supervisores o calidad realicen alguna consulta cerrará la hora de término con la misma hora de inicio y que indique 0:0 el tiempo de la actividad.
- Nueva Orden de Trabajo
    - Apartado auxiliar para dar de alta las O.T.
    - Los datos que se registrarán son los siguientes: número de O.T., pedido y partida, cliente y aplicación (producto)
    - Aquí se darán de alta o baja los equipos de producción.
    - Para identificar rápidamente las máquinas, se seleccionarán de una lista desplegable después de elegir el área correspondiente.
  - Consulta de Producción
    - Accesible sólo para el área de calidad y jefes de producción donde podrán hacer la consulta de tiempos de producción y actividades que involucran a las O.T., producto, máquinas y operadores.
    - Opción para descargar en formato XLS la consulta realizada.

- Edición de Reportes
  - Permite realizar correcciones en las actividades de los usuarios si así lo requieren. Sólo el área de calidad tendrá acceso a este apartado.
  - Después de localizar el reporte, mostrará los campos que podrán ser editables.
  - Al guardar los cambios el sistema volverá a hacer el cálculo para obtener el nuevo tiempo total de la actividad.
- Registro de nuevos usuarios
  - Se darán de alta las nuevas personas que entren a trabajar a la empresa y se darán de baja cuando salgan de la organización.
  - Seleccionará el nivel de privilegios dentro del sistema.
- Se tiene que ocupar el software y hardware que ya tiene la empresa para el desarrollo e implementación de SiRPAG.

### c. Características de los usuarios

En la tabla 3.3 se muestra el desglose de los privilegios de acceso a los componentes del sistema solicitados en SiRPAG:

Tabla 3.3 Nivel de privilegios

<b>Tipo de Usuario</b>	<b>Función(es)</b>	<b>Nivel</b>	<b>Actividades en SiRPAG</b>
Operador	Máquinas de impresión, acabado, acabado manual o empaque.	C	Son todos aquellos que únicamente emplearan la pestaña <i>Actualizar Actividad</i> para registrar su trabajo diario.
Jefes y supervisores de producción	Control, máquinas de impresión, acabado, acabado manual o empaque.	B	Además de <i>Actualizar Actividad</i> , tendrá la pestaña <i>Nueva O.T.</i> y <i>Consulta de Producción</i> habilitados para su uso.
Área de Calidad	Calidad, reportes de status y comportamiento de producción.	A	<i>Actualizar Actividad</i> , <i>Nueva O.T.</i> <i>Consulta de Producción</i> y <i>Edición de Reportes</i> habilitados para su uso.
Recursos Humanos	Administración, gestión de nóminas y beneficios del personal.	S	<i>Registro de Usuarios</i> únicamente.

#### d. Restricciones y Evolución Estimada del Cliente

**Restricciones del software.** La organización cuenta con licencia para desarrollar en Visual FoxPro sus aplicaciones y procesos de datos dentro del área de dato variable, un software descontinuado pero que ha suplido los requerimientos de los clientes en la actualidad.

El proyecto de innovación en sus sistemas informáticos inició recién, actualmente es requerimiento primordial contar con un sistema de captura de reportes desarrollado con el software que posee la empresa.

Cuando determine la empresa la licencia de un nuevo software para desarrollo de aplicaciones se piensa migrar SiRPAG contando con una estructura similar, pero en un entorno nuevo.

La metodología de desarrollo será en cascada modificada, por razones de tiempo y disponibilidad se dejará sentado los elementos necesarios para el desarrollo del producto sin interrupciones, retrocesos, cambios que impacten en el tiempo y acordando así una fecha límite.

**Restricciones del hardware.** La aplicación y base de datos estarán alojadas en un servidor Microsoft Windows Server 2012 destinado a este tipo de programas. Se usarán equipos de cómputo que ya posee la Empresa:

- PC con procesador Intel Pentium D
- Sistema Operativo Microsoft Windows 7 Home Premium 32 bits
- Memoria RAM de 2 GB
- Disco duro de 320 GB
- Pantalla VGA de 1024 x 768 y 1366 x 768 px
- Mouse y teclado
- No Break

#### 3.1.3 Requerimientos del desarrollador

##### **Requerimiento #01 Comunicación entre los principales sistemas de información**

- El nuevo Sistema de Cotizaciones Mecano tendrá una tabla en su base de datos con los campos orden, pedido, partida, cliente y producto que permitirá a SiRPAG consultar los números de O.T. que no tenga dentro de la tabla OT (una de las tablas que compone a la base de datos que utilizará SiRPAG) y poder darla de alta de manera automática.
- Para establecer esta comunicación cada equipo tiene que tener la conexión habilitada en el Administrador de orígenes de datos ODBC<sup>4</sup>.
  - El encargado de desarrollo del sistema de cotizaciones le proveerá de la información requerida para dar de alta la conexión:
    - Dirección IP donde se localiza la base de datos de Mecano.
    - Usuario y contraseña de autenticación de la base de datos.
    - Nombre de la base de datos
    - Nombre de la tabla (esto para programar la comunicación en SiRPAG)
- Cada vez que un usuario registre una actividad SiRPAG buscará primeramente en su tabla OT la O.T., si no la localiza buscará en la tabla compartida con el Sistema Mecano.

<sup>4</sup>Open Database Connectivity (conectividad abierta de base de datos).

**Requerimiento #02 Ventana inicial que solicite al usuario su nombre y contraseña para autenticar que sea un empleado de la empresa e inicie sesión.**

- Una ventana que el lado izquierdo contenga los elementos de área de trabajo, nombre del usuario, contraseña, iniciar sesión y la opción de cambiar contraseña.
- Área de trabajo será una lista desplegable con información obtenida de la tabla USUARIO. Al seleccionar algún registro se alimentará la siguiente lista de Nombre de Usuario con los usuarios que pertenecen al área de trabajo seleccionado.
- Al seleccionar el área y el nombre del usuario sólo tendrá que escribir la contraseña, si se equivoca al escribir la contraseña se cierre después del quinto intento.
- Un botón de salida y a la derecha una portada con el nombre de la empresa.
- Botón que los lleve a una ventana donde puedan consultar su historial personal.
- Acceso a un manual de usuarios cuando así lo requieran en PDF.

**Requerimiento #03 Una opción para poder hacer cambio de su contraseña y consultar su historial.**

- Ventana auxiliar que aparecerá al seleccionar el cambio de contraseña y recibirá la nueva contraseña de máximo 15 caracteres alfanuméricos.
- La persona puede guardar el cambio o cancelar y volver al inicio.
- Para poder llamar el historial es necesario que el usuario escriba correctamente su contraseña, al quinto error se cierre el sistema.
- Tres botones para hacer consulta, descarga e impresión, y una breve descripción de cómo realizar la consulta.
- Para poder consultar debe haber dos botones que muestren un calendario y seleccionar el día inicial y el día final como rango de tiempo.
- Que aparezca una tabla con los resultados de la consulta a detalle.
- Para descargar los resultados el usuario tendrá que dar clic en el botón de descargar y seleccionar la ruta donde se descargará la consulta en formato XLS.
- Para poder visualizar el archivo se utilizará el software libre Apache OpenOffice.
- Con el botón Imprimir mostrará la ventana previa para elegir la impresora.

**Requerimiento #04 Contener una pestaña con la opción de Actualizar Actividades que estén relacionadas a una O.T. o adicionales, cada que genere una nueva actividad tendrá que cerrar la anterior excepto para Impresión Variable y Rígidos quienes usan más de un equipo a la vez. Habrá campos que se llenarán de forma automática y si el usuario no cierra su reporte no será considerado su tiempo de producción y tendrá que solicitar el cambio a Calidad.**

- Un cuadro de texto donde llenará el número de O.T. la cual cargará el usuario su actividad, si no existe la O.T. no puede reportar la actividad.
- Un botón para hacer la búsqueda, primero en la base de datos de SiRPAG en la tabla OT, si no la localiza allí se comunicará a la tabla compartida con el nuevo Sistema de Cotizaciones Mecano en desarrollo.
- Al localizar la O.T. habilitará los campos: Nueva Actividad, Área, Máquina y Comentarios.
- Área y Máquina tendrán listas desplegables que mostrarán información de la tabla MAQUINA.
- Nueva Actividad tendrá una lista que desplegará la información de la tabla CLAVE que son las diversas actividades que se ejecutan en la empresa:

- Preparar máquina
  - Arreglo de máquina
  - VoBo
  - Producción
  - Secado
  - Espera de insumos
  - Mantenimiento preventivo
  - Mantenimiento correctivo
  - Capacitación
  - Acabado y/o apoyo otras actividades
    - Engomado
    - Clasificación
    - Empaque
    - Engrapado
    - Recibir en máquina
    - Sellado manual
    - Embolsado manual
    - Deshojado
    - Etiquetado
  - Reoperaciones
  - Recuperaciones
  - Problemas
  - Ripeo de archivo
  - Preparación de tinta
- Serán estos tres campos obligatorios que llenar para poder guardar el reporte de actividades. Excepto las actividades que no posean número de O.T. que se conocen como actividades fuera de producción, sólo tendrán el campo de Nueva Actividad como obligatorio.
  - Por lo tanto, existen dos tipos de reportes, unos con O.T. asignada y otros no.
  - Al seleccionar la actividad de “Acabado y/o apoyo otras actividades” se habilitará un listado con las actividades específicas de este tipo, el listado no requiere ser alimentado por ninguna tabla, ya que son procesos genéricos y limitados.
  - El área será una lista desplegable con la información de las diferentes áreas a las que pertenecen las máquinas de la compañía. Al seleccionar una, ésta alimentará la siguiente lista de máquinas pertenecientes.
  - La lista de máquinas sólo mostrará las que pertenezcan al área seleccionada.
  - En la caja de comentarios se llenarán datos adicionales y no es obligatoria para guardar el reporte de actividades.
  - Después de seleccionar alguna máquina del área de rígidos se mostrarán en pantalla los campos de largo, ancho (con éstos se calcula el área del material) y tipo de material (listado de los materiales que trabajan las máquinas del área).
  - Al seleccionar el botón Guardar se almacenarán los datos en la tabla REPORTE: O.T., aplicación, operador, fecha, máquina, actividad, hora Inicial y comentarios.
  - Cuando el operador acabe su actividad buscará la O.T. y se mostrará la actividad y máquina que había registrado, si las actividades que seleccionó fueron: producción, embolsado manual, empaque, acabado y/o apoyo otras actividades o recuperaciones

se visualizará el título Cantidad y a su lado un cuadro de texto para que el operador escriba el número de producto terminado.

- A la derecha del cuadro de cantidad deberá aparecer la unidad de medida del producto dependiendo del tipo de máquina, algunas se manejan por extendido, revolución, pieza, producción completa, páginas, paquetes o contadores. Para determinar la unidad de medida se les asignarán banderas dentro de la tabla MAQUINA.
  - En la mitad inferior de la ventana se visualizarán los mismos elementos de registro de actividad para que el usuario si así lo requiere, y registre una nueva actividad perteneciente a la misma O.T.
  - Si ha terminado de registrar la conclusión de su actividad al dar clic en guardar almacenará la hora de término, la cantidad del producto terminado, la actualización de los comentarios y el total de horas y minutos que tomó la actividad en concluirse.
- Hay un caso especial con dos áreas: impresión variable y rígidos. Los operadores de estas áreas llegan a usar hasta tres equipos a la vez por lo que debe haber dos botones visibles a la hora de registrar actividades.
    - Un botón es cerrar máquina y el otro registrar máquina.
    - Cuando el usuario guarde una actividad con una máquina asignada a la O.T. y desea registrar otra actividad, pero usando otra máquina para la misma O.T. se visualizarán estos botones para dar de alta y guardar las actividades.
  - Al buscar nuevamente la O.T. para finalizar la tarea al dar clic en cerrar máquina se visualizará una lista desplegable de las máquinas que tiene dadas de alta, al seleccionar alguna mostrará los datos guardados para que al dar clic en Guardar se asigne la hora de término y cantidad a esa máquina.
  - Un elemento adicional al cerrar una actividad del área de Rígidos es que se habilitará el campo de número de pasadas (se refiere al número de pruebas)
  - Si la actividad es Preparación de tinta en el Área de Rígidos se habilitarán al cerrar la actividad los campos de Cyan, Magenta, Amarillo, Negro, Light Magenta y Light Cyan que son las tintas que poseen los equipos de impresión del área que al ser recargados se tienen que reportar las cantidades.

**Requerimiento #05 Contener una pestaña con la opción de Nueva Orden de Trabajo que permita dar de alta las O.T. de manera manual y dar de alta y baja las máquinas de la Empresa.**

- La pestaña cuenta con tres botones: Nueva O.T., Guardar y Cancelar.
- Al hacer clic a Nueva O.T. se visualizarán los siguientes elementos:
  - La palabra O.T. y a continuación un cuadro de texto que aceptará únicamente números para la nueva O.T.
  - La palabra Pedido y un cuadro de texto que acepte sólo números.
  - La palabra Partida y un cuadro de texto que acepte sólo números.
  - Debajo de la O.T. la palabra Cliente y una lista desplegable a la derecha que mostrará los nombres de los clientes que existen en la tabla OT.
  - A la derecha de la lista un botón que remplace la lista por un cuadro de texto donde se escribirá el nuevo cliente.
  - Debajo de Cliente la palabra Aplicación, a su derecha una lista desplegable que mostrará los productos registrados en la tabla OT que pertenecen al cliente seleccionado en la lista anterior.

- A la derecha de la lista un botón que reemplace la lista por un cuadro de texto donde se escribirá el nuevo producto.
  - Debajo de Aplicación la palabra Área y a su derecha la lista desplegable de áreas operativas de la tabla MAQUINA.
  - Debajo de Área la palabra Máquina y a su derecha una lista desplegable de máquinas que pertenecen al área seleccionada de la lista anterior.
  - A la derecha de la lista un botón para agregar una nueva máquina que reemplazará la lista por un cuadro de texto donde se ingresará el nuevo elemento, otros elementos que se visualizarán en la parte de abajo son los siguientes:
    - La palabra Marca y un cuadro de texto a la derecha.
    - La palabra Tinta y un cuadro de texto acepte números.
    - La palabra Proceso y un cuadro de texto a la derecha.
    - La palabra Operadores y un cuadro de texto que acepte números.
    - La palabra Unidad de medida y una lista desplegable con los elementos: extendido, revolución, pieza, producción completa, páginas, paquetes y contadores.
      - Al guardar la nueva máquina almacenará la bandera que pertenece a la unidad de medida seleccionada.
  - Al dar clic en el botón nuevamente, guardará el registro con sus datos de marca, tinta, proceso, operadores y unidad de medida y volverá a visualizar la lista ya actualizada.
  - A la derecha del botón para agregar una nueva máquina estará el botón de Eliminar máquina. Después de seleccionar en la lista el equipo se le dará clic a este botón que eliminará el registro.
- Al hacer clic en el botón Guardar se almacenará la información en la tabla OT.
  - Al hacer clic en Cancelar oculta los elementos mostrados por el botón Nueva O.T.

**Requerimiento #06 Contener una pestaña con la opción de Consulta de Producción que permita mostrar y descargar los tiempos de producción por O.T., producto, máquina y operador.**

- La pestaña cuenta con dos grupos de botones, el primero es el Tipo de Consulta: producto, máquina, operador y O.T. El segundo es el Archivo: generar y descargar.
- El botón Producto visualiza el texto Cliente y una lista desplegable de los clientes existentes en la tabla OT.
- Debajo de Cliente mostrará el texto Producto y una lista desplegable de los productos que pertenecen al cliente seleccionado.
- A la derecha está el texto día y un botón de un calendario dos veces, permitirán seleccionar el rango de fechas de la consulta.
- El botón Máquina visualiza el texto Área y una lista desplegable de las áreas operativas de la tabla MAQUINA.
- Debajo de Área mostrará el texto Máquina y una lista desplegable de las máquinas que pertenecen al área seleccionada.
- Los dos calendarios permitirán seleccionar el rango de fechas de la consulta.
- El botón Operador visualiza el texto Área y una lista desplegable de las áreas de trabajo registradas en la tabla USUARIO.

- Debajo de Área mostrará el texto Operador y una lista desplegable de los usuarios que pertenecen al área seleccionada.
- A la derecha está el texto día y un botón de un calendario dos veces, permitirán seleccionar el rango de fechas de la consulta.
- El botón Orden de Trabajo visualiza el texto con el mismo nombre y un cuadro de texto que aceptará sólo números para consultar la O.T.
- Tras elegir la consulta, se seleccionará el botón Generar, éste mostrará una visualización de la consulta en la ventana.
- Con el botón Descargar mostrará una ventana donde el usuario elegirá donde se guardará la consulta en archivo XLS. Al guardar se limpiará la ventana.

**Requerimiento #07 Contener una pestaña con la opción de Edición de Reportes que permita corregir los reportes creados dentro del sistema o eliminarlos con sólo dar los datos del número de O.T., usuario, máquina, fecha, actividad y hora inicial.**

- La pestaña cuenta con los botones Guardar cambios, Eliminar y Cancelar.
- En la parte superior de la ventana se mostrará el texto O.T. y un cuadro de texto que aceptará sólo números.
- Un botón a la derecha de búsqueda que visualizará los siguientes campos:
  - Texto Usuario y una lista desplegable con los usuarios que han reportado sobre esa O.T.
  - Texto Máquina y una lista desplegable con las máquinas que ha reportado el usuario seleccionado.
  - Texto Fecha y una lista desplegable con los días que ha reportado el usuario con la máquina seleccionada.
  - Texto Actividad y una lista desplegable con las actividades realizadas el día seleccionado.
  - Texto Hora Inicial y las horas en que se realizaron la actividad y fecha seleccionadas.
- Si el reporte no existe no mostrará nada, si existe y se seleccionan todos los campos al volver a dar clic en la búsqueda mostrará en la parte inferior de la ventana los siguientes elementos:
  - Área y Máquina con sus listas desplegables para actualizar la información.
  - Hora Inicial y Hora Final con sus cuadros de texto para poder editar los tiempos.
  - Cantidad y cuadro de texto que sólo acepte números.
- Para guardar los cambios se dará clic en el botón Guardar Cambios, al hacer eso reemplazará la información del reporte seleccionado.
- Para eliminar ese reporte se dará clic en el botón Eliminar después de dar clic en la búsqueda al seleccionar los campos iniciales.
- El botón Cancelar inhabilitará los campos y limpiará la pantalla para hacer una nueva búsqueda.

**Requerimiento #08 Contener una pestaña con la opción de Registro de Nuevos Usuarios que permitirán dar de alta y de baja los usuarios de SiRPAG.**

- La pestaña cuenta con los botones Nuevo usuario, Quitar usuario, Guardar y Cancelar.
- Al seleccionar el botón Nuevo Usuario se visualizarán los siguientes campos:
  - Texto Número de Usuario (que fungirá como contraseña) y un cuadro de texto para sólo números, esto es para que se den de alta en la tabla USUARIO.
  - Texto Nombre de Usuario y un cuadro de texto que acepte sólo mayúsculas para ingresar el nombre completo del nuevo usuario.
  - Texto Área y una lista desplegable con la lista de áreas de trabajo que cuenta la empresa.
  - Texto Nivel y una lista desplegable con las letras: A, B, C y S.
    - A: Personal del área de Calidad
    - B: Gerentes, jefes y supervisores
    - C: Operadores
    - S: Recursos Humanos
- Con el botón Guardar se almacenará la información escrita.
- El botón Cancelar limpiará la pantalla de datos ingresados y elementos habilitados.
- Con el botón Quitar Usuario se visualizarán los siguientes elementos: texto Nombre del Usuario, una lista desplegable con todos los miembros de la empresa registrados en SiRPAG y un botón a la derecha.
- Después de seleccionar algún nombre de la lista, con el botón de la derecha se eliminará el registro de la tabla USUARIO.

**Requerimiento #09 Se tiene que ocupar el software y hardware que ya tiene la empresa para el desarrollo e implementación de SiRPAG. Usando Visual FoxPro 9.0 en el software y diversos PC que hay en la empresa.**

- Los equipos de cómputo que usarán SiRPAG deben tener almacenados e instalados los DLL para ejecutar aplicaciones de FoxPro.
- Las pantallas de los equipos deben tolerar una dimensión superior a 1024 x 640 px.
- El estilo de la interfaz tiene que ser similar a Windows, la pantalla de inicio tiene que verse como lo muestra la figura 3.2

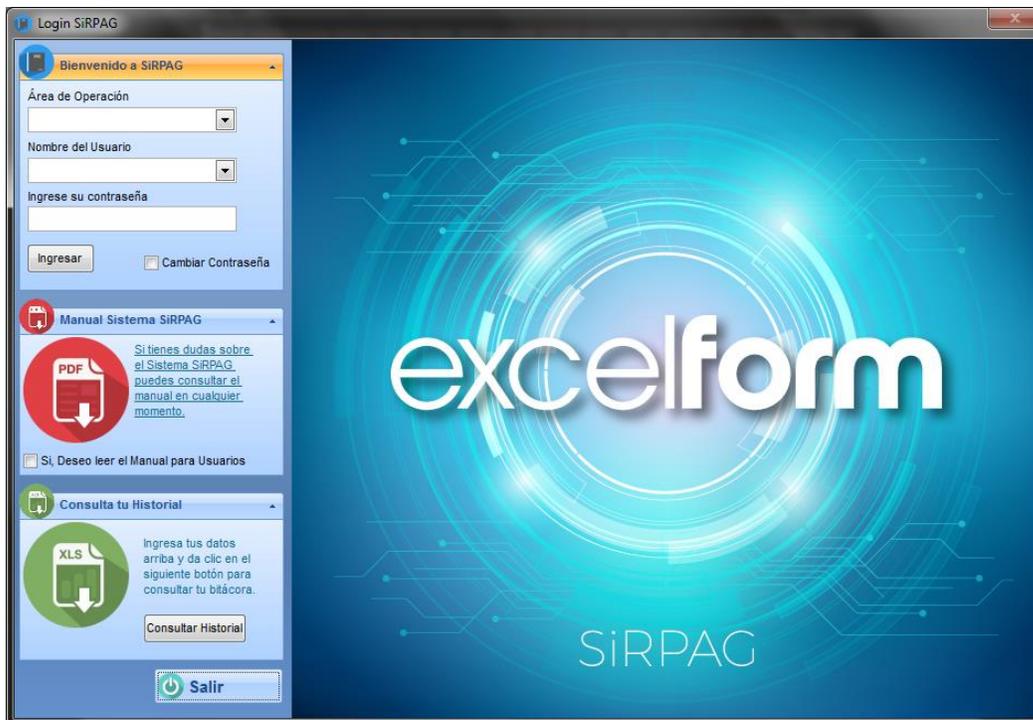


Figura 3.2 Vista preliminar del Inicio de Sesión de SiRPAG.

- Instalar Adobe PDF en los equipos para poder visualizar el manual de usuarios.
- Instalar Apache OpenOffice que permita visualizar los archivos XLS generados.
- El sistema debe estar alojado en el Servidor Windows Server 2012, se tiene que generar el ejecutable en el mismo servidor.
- Los equipos estarán conectados a la unidad de red del Servidor.
- En el escritorio de los equipos que utilizarán los usuarios tendrán el acceso directo al ejecutable.

**Requerimiento #10 Según sea la pestaña de SiRPAG sólo personal autorizado tendrá acceso a ella a través de una clasificación de letras: C, B, A y S.**

- En la parte superior tendrá atajos para acceder al historial personal, volver a la ventana de Inicio cerrando la sesión del usuario, cambiar contraseña y Ver el manual.
- No podrá el usuario minimizar la ventana para evitar dejar su sesión abierta después de acabar sus actividades dentro del sistema.
- Con dar clic en el botón de cerrar en la esquina superior derecha saldrán de la aplicación y cerrará la sesión del usuario.
- Debajo de los atajos deben estar las pestañas para cambiar a las opciones deseadas: Actualizar Actividad, Nueva O.T., Consulta de Producción, Edición de Reportes y Registro de Usuarios.
- Al seleccionar alguna opción ésta volverá visible el contenido que se requiere.
- En la parte inferior de la ventana debe decir el nombre del usuario que tiene la sesión abierta para que pueda identificar en todo momento que la ventana le pertenece.
- Los botones principales tienen que tener iconos con imágenes que den referencia a lo que hacen y palabras clave de lo que hacen, ejemplo:
  - Botón de Búsqueda con la imagen de una lupa.

- Botón de Guardar con una paloma de aceptación de color verde.
  - Botón de Cancelar con la imagen de un tache color rojo.
  - Todos los botones que se repitan a lo largo de las pestañas que contengan imágenes iguales para que se familiarice el usuario rápidamente.
- El tamaño de la letra sea legible, no menos de Arial tamaño ocho.
  - Respetar el nivel de privilegios mencionado en el apartado de características de los usuarios.

## 3.2. Diseño del sistema

### 3.2.1 Diseño preliminar

En los requerimientos funcionales se mencionan las configuraciones que deben presentarse a la hora de implementar SiRPAG. La figura 3.3 nos transmite que, para poder localizar una nueva O.T. dentro de SiRPAG si no lo encuentra en la tabla OT del mismo, accede a través del ODBC a la base de datos del nuevo sistema de cotizaciones para buscar en la tabla correspondiente la O.T. que desea el usuario. Si la O.T. no es encontrada entonces no se podrá generar un reporte de producción de la misma.

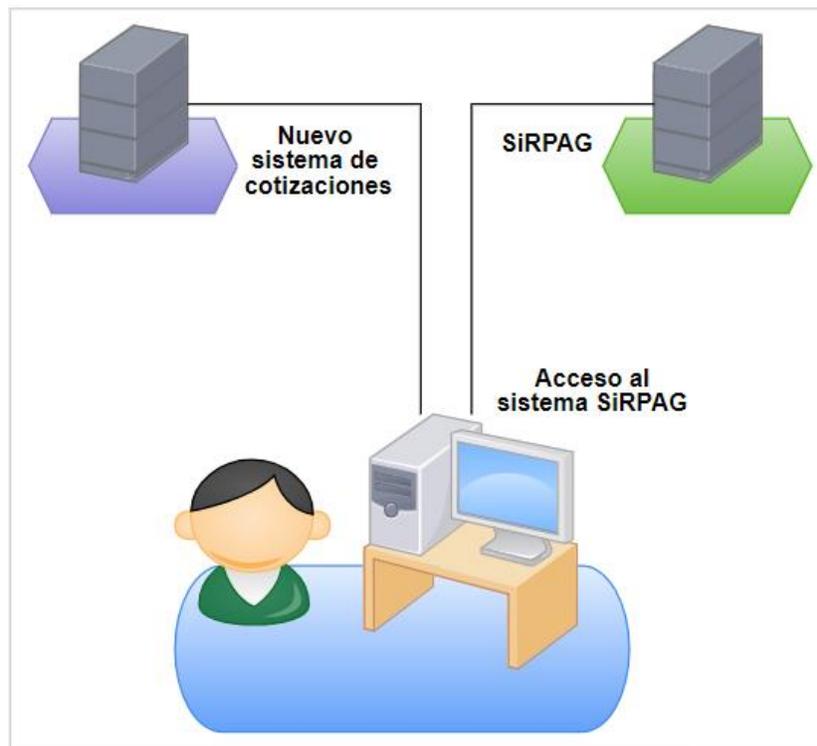


Figura 3.3 Arquitectura de software de SiRPAG.

Si la O.T. existe, pero en el sistema anterior de cotizaciones y pedidos al cual no es posible acceder a su base de datos, SiRPAG contará con un apartado especial para dar de alta de manera manual la O.T. Este apartado no es visible para los operadores, sólo para quienes tienen acceso. Los módulos que cuenta SiRPAG podrán ser accedidos a partir del nivel de privilegios que posea el usuario como se muestra en la figura 3.4.

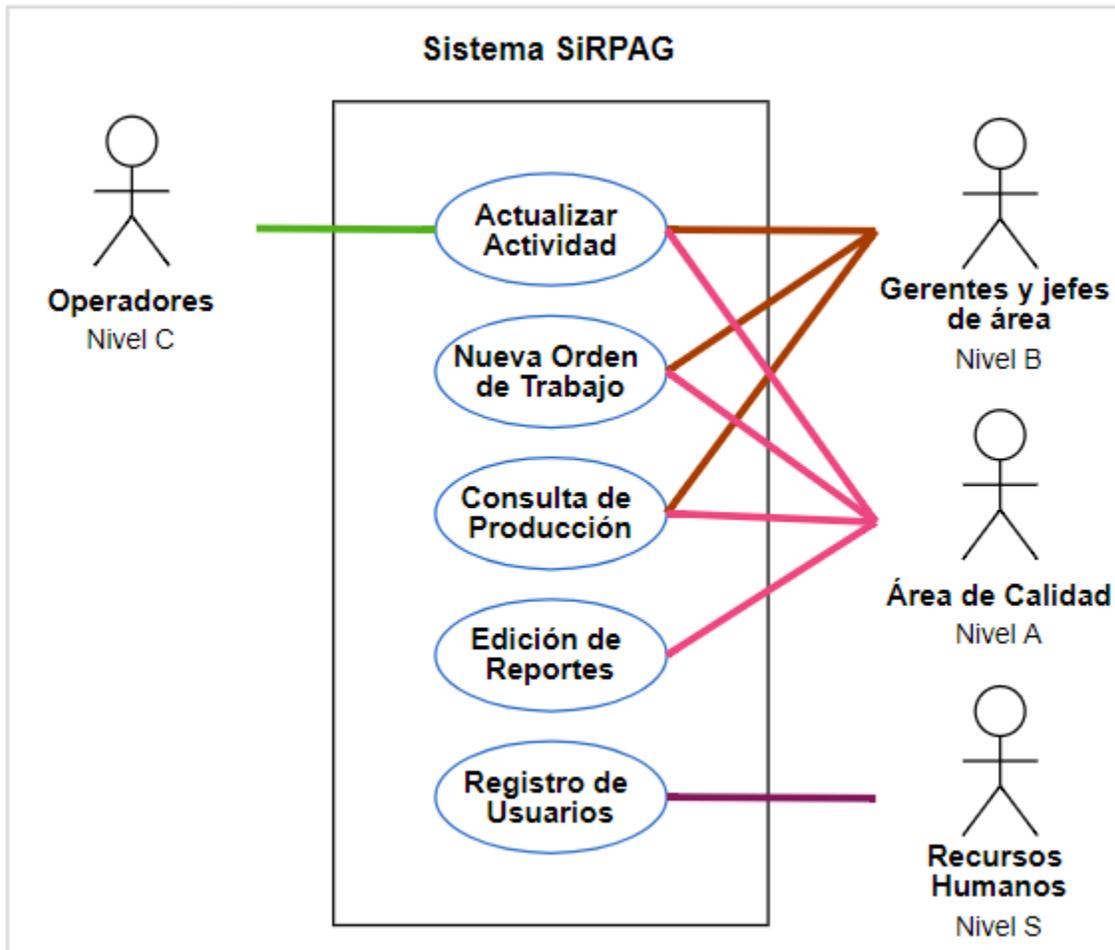


Figura 3.4 Nivel de privilegios en SiRPAG caso de uso.

Los elementos genéricos y que todo usuario puede ver en SiRPAG son: inicio de sesión, cambio de contraseña, historial del usuario, manual operativo y la ventana inicial del sistema de reportes.

Los usuarios de nivel C sólo verán en la ventana del sistema el apartado de Actualizar Actividad. Los usuarios de nivel B verán como los usuarios de nivel C el apartado de Actualizar Actividad, pero podrán acceder a Nueva Orden de Trabajo y Consulta de Producción. El nivel A permite los privilegios del nivel B más el acceso al apartado Edición de Reportes. Por último, el nivel S sólo podrá usar el apartado de Registro de Usuarios.

El cambio de entorno no puede ser drástico, y debe haber campos que se obtengan de manera automática sin intervención del usuario, partiremos mostrando las actividades generales que podrá hacer el usuario dentro de SiRPAG a través del diagrama de actividades de la figura 3.5.

### **Ventana Inicial**

En la pantalla inicial solicita un nombre y una contraseña, el primero se elige de una lista desplegable, tras el llenado de la información se pueden realizar las siguientes acciones:

- Cambio de contraseña: al salir regresa al Inicio de Sesión.
- Consultar el historial del usuario: te permite elegir regresar a la pantalla de inicio o entrar al sistema administrador.
- Entrar al sistema SiRPAG

### **Ventana SiRPAG**

En la parte superior están las pestañas habilitadas con los procesos que se pueden hacer conforme al nivel de privilegios que posea el usuario. Las actividades son las siguientes:

- Actualizar Actividad: se busca la O.T. dentro de la base de datos del sistema, si no la encuentra busca en el nuevo sistema de cotizaciones, al localizarlo habilita los campos para guardar los cambios y actualizar el estatus de producción.
- Registrar O.T.: opción para registrar de manera manual, si aparecen en las listas desplegadas los productos, se puede habilitar la opción de escribirlos o permite dar de alta máquinas en la base de datos y actualizar la información.
- Consulta de Producción: genera la consulta con la opción de elegir por producto, máquina, operador u orden de trabajo, con la opción de descargar la consulta en XLS.
- Edición de Reportes: busca el reporte a partir de la O.T. y otros elementos, permitiendo eliminar el reporte o modificarlo.
- Nuevo Usuario: registra un nuevo usuario o elimina alguno existente.



### 3.2.2 Diseño detallado

#### Estructura de datos

- **Modelo de flujo de datos**

#### Nivel de contexto:

El diagrama que se muestra en la figura 3.6 es el estado general del sistema donde participan las diferentes entidades que trabajan con SiRPAG.

*Entidad externa Cliente:* representa a cualquier usuario registrado en el Sistema, el cual dependiendo del nivel de privilegios podrá hacer una o varias de las acciones a partir de los datos de entrada que introduzca en el programa.

*Entidad externa Mecano:* la relación del nuevo sistema de cotizaciones con SiRPAG es de consulta de las O.T. con su respectiva información.

*Entidad Externa XLS:* los archivos XLS son el resultado de las consultas de producción.

*Proceso SiRPAG:* es el sistema administrador de la producción, gestiona todo lo relacionado a los reportes de producción y trabajos sobre las máquinas de la empresa, así como las diversas actividades que hacen día a día los usuarios.

*Flujo de datos Actualizar reportes:* son los datos que se darán de alta o actualizarán a partir del número de O.T. o seleccionando alguna actividad sin orden asignada.

*Flujo de datos Ingresar nuevos elementos:* representa la información de las máquinas de la empresa y/o alguna O.T. que no pertenezca al Mecano.

*Flujo de datos Consultar producción:* es la información que se ha registrado a lo largo del intervalo de fechas y del tipo de consulta seleccionados.

*Flujo de datos Editar reportes de producción:* son los datos que sustituirán a los de algún reporte del sistema o que se eliminarán de la tabla REPORTE.

*Flujo de datos Registro de usuarios:* es la información del usuario que será registrado en el sistema o del que será eliminado.

*Flujo de datos Busca O.T. en Mecano:* resultado de la consulta por DSN con Mecano para localizar la información necesaria para dar de alta reportes de producción.

*Flujo de datos Respuesta de la búsqueda en Mecano:* como su nombre lo indica es la información respuesta a la consulta a la Entidad externa Mecano.

*Flujo de datos Archivo de consulta:* es la información que se obtuvo a la hora de generar la consulta, éstas se guardarán en un archivo XLS en la PC.

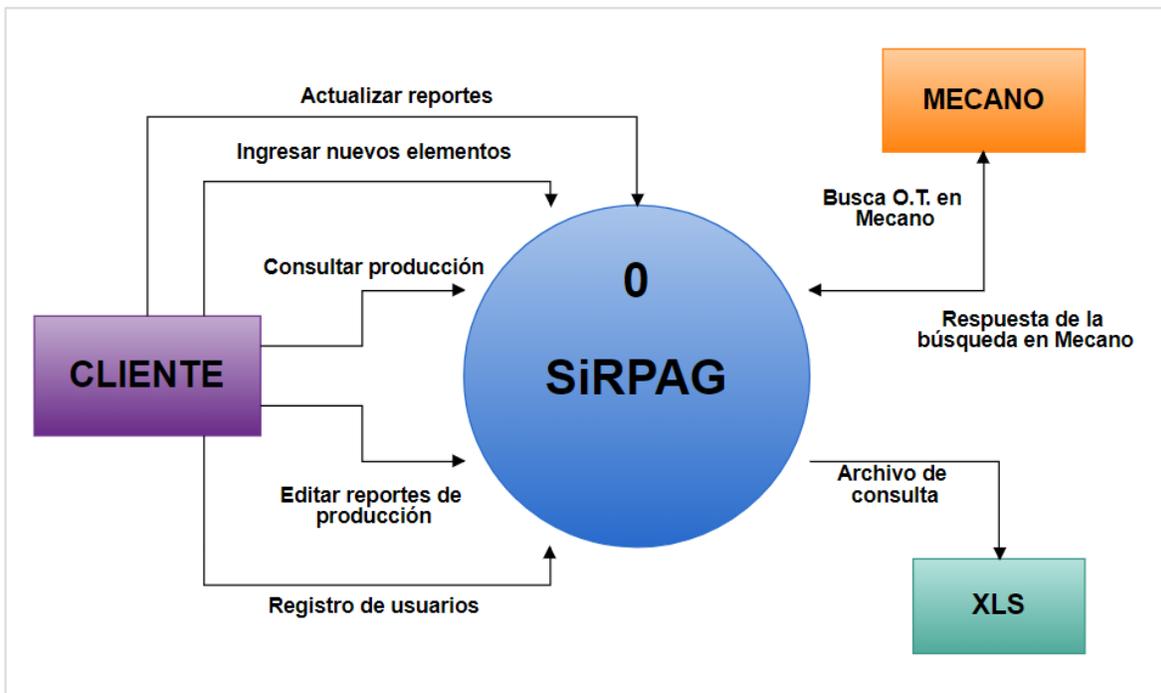


Figura 3.6 Diagrama de contexto del sistema SiRPAG.

#### Nivel 0:

Al accionar algún proceso disponible en el sistema se opera con una serie de instrucciones que en el caso del Proceso 3: registrar reportes se describen en la figura 3.7 de forma detallada.

*Proceso Inicio de sesión:* el nombre de usuario y contraseña escritos son procesados y comparados con la tabla USUARIO para validar si es correcta o no la información.

*Proceso Consultar historial:* recibe el rango de fechas para que el usuario pueda solicitar una vista previa de su historial personal a través de una consulta en la tabla REPORTE.

*Proceso Registrar reportes:* este proceso se descompone a su vez en subprocesos, en términos generales recibe un número de O.T. o una actividad sin O.T. que permitirán registrar y actualizar los reportes laborales del usuario.

*Proceso Máquinas, productos y nuevas O.T.:* registra una nueva O.T. en la tabla OT, adicional permite dar de alta o de baja información en la tabla MAQUINA.

*Proceso Tipo de consulta:* a partir del intervalo de fechas y el tipo de consulta (por producto, máquina, operador u O.T.) se crea la vista previa de la consulta.

*Proceso Buscar actividad:* se tienen que elegir los datos que permitirán hacer la búsqueda del reporte que podrá o ser eliminada o editada su información.

*Proceso Altas y bajas:* introduce los datos para registrar un nuevo usuario o eliminarlo.

*Flujo de datos Perfil de usuario:* el nombre y la contraseña para iniciar sesión.

*Flujo de datos Fecha inicial y final:* rango de fechas que requiere el sistema para poder realizar la consulta del historial del operador.

*Flujo de datos Cambiar contraseña:* representa los cambios en la tabla USUARIO.

*Flujo de datos Administrador en actualizar reportes:* representa el número de O.T. o el número asociado a las actividades extras no asignadas a una orden.

*Flujo de datos O.T. / actividad actualizada:* son los datos que se registran dentro de la tabla BITACORA para tener un historial de las actividades ejecutadas dentro de SiRPAG.

*Flujo de datos Administrador en ingresar nuevos elementos:* representa los datos que serán dados de alta para generar una nueva O.T. y editar máquinas.

*Flujo de datos O.T. nueva:* es la información para la tabla BITACORA sobre la nueva O.T.

*Flujo de datos Nueva máquina:* son los datos para dar de alta una nueva máquina de la empresa en la tabla MAQUINA.

*Flujo de datos Busca eliminar máquina:* es la información solicitada a la tabla MAQUINA de los equipos de la empresa que se deseen dar de baja.

*Flujo de datos Respuesta de eliminar máquina:* es la información actualizada después de haber sido dada de baja la máquina de la tabla MAQUINA.

*Flujo de datos Alta de O.T.:* son los datos para poder dar de alta una nueva O.T. en la tabla OT y poder ser asignadas en los reportes de producción.

*Flujo de datos Administrador en consultar producción:* representa la información que desea ser consultada en la tabla REPORTE.

*Flujo de datos Resultados de consulta:* es la información del tipo de consulta que se realizó, acción que se reporta a la tabla BITACORA.

*Flujo de datos Administrador en editar reportes de producción:* es el número de O.T. para localizar el reporte de la tabla REPORTE.

*Flujo de datos Pregunta reporte a cambiar o eliminar:* representa la información para localizar el reporte para poder ser editado o eliminado.

*Flujo de datos Respuesta del reporte a cambiar o eliminar:* son los datos que podrán ser editados o eliminados de la tabla REPORTE.

*Flujo de datos O.T. / actividad editada:* es la información para dar de alta en la tabla BITACORA sobre los cambios que se realizaron en el reporte.

*Flujo de datos Administrador en registro de usuarios:* representa los datos que serán dados de alta o baja en la tabla USUARIO.

*Flujo de datos Busca eliminar usuario:* es la información solicitada a la tabla USUARIO sobre los registros que se quieran dar de baja.

*Flujo de datos Respuesta de eliminar usuario:* son los registros que están actualmente en la tabla de USUARIO.

*Flujo de datos Dar de alta usuario:* es la información que se da de alta en la tabla USUARIO.

*Flujo de datos Cambio en usuarios:* es la información de las acciones del usuario para la tabla BITACORA.

*Almacén Usuario:* representa la tabla USUARIO que contiene la información de los usuarios registrados en el sistema.

*Almacén Máquina:* representa a la tabla MAQUINA que contiene la información de todos los equipos de producción.

*Almacén OT:* representa a la tabla OT que posee la información de las O.T. registradas dentro del sistema y las que obtenidas a través de Mecano.

*Almacén Reporte:* representa la información de todos los reportes ingresados.

*Almacén Bitácora:* representa a la tabla BITACORA que guarda la información de cada cambio o movimiento en las tablas USUARIO, OT y REPORTE.

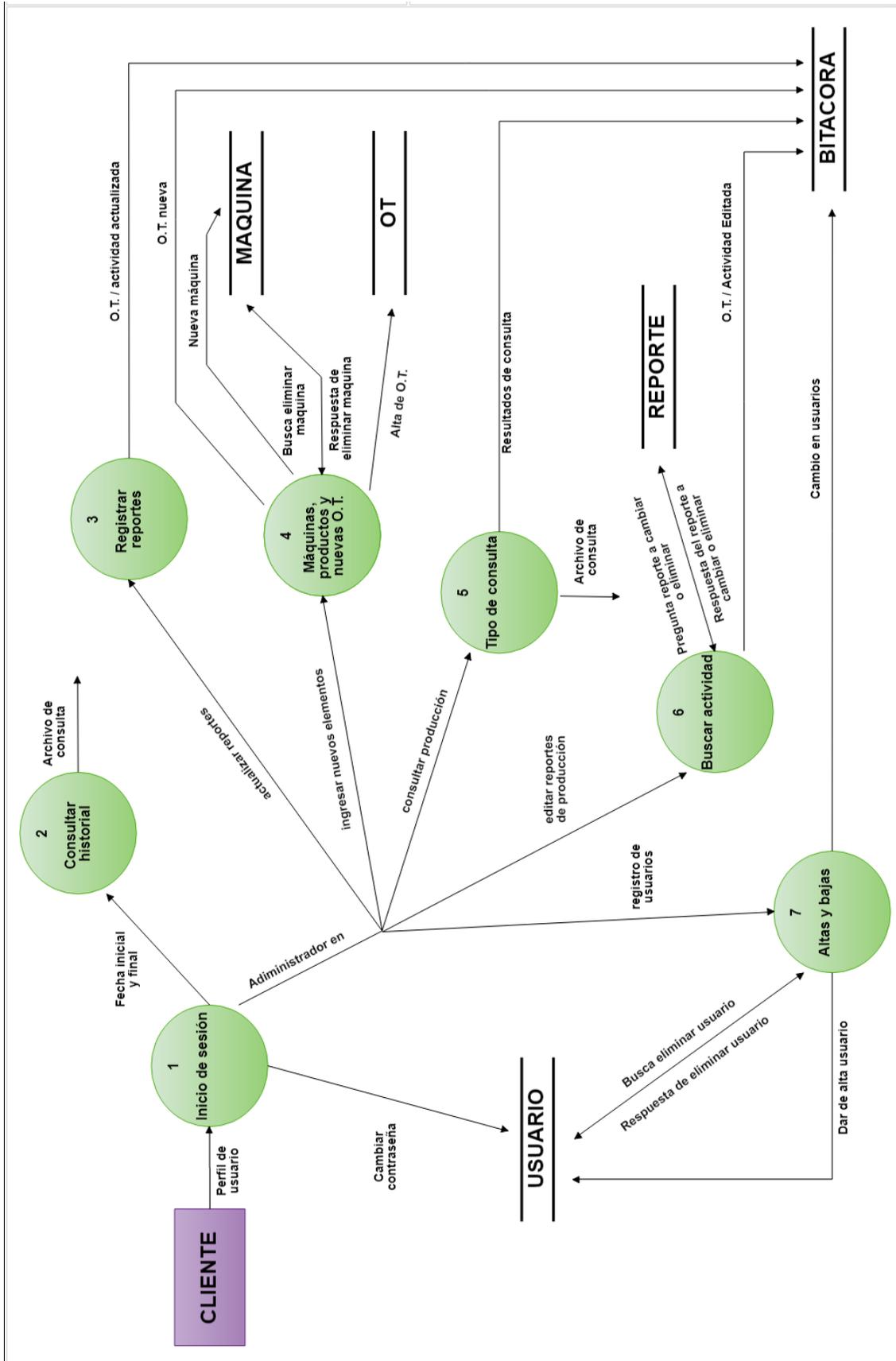


Figura 3.7 DFD Diagrama 0: Opciones del sistema.

## Nivel 1:

*Proceso Tipo de actividad:* tiene dos funciones, la primera es que al recibir un número de O.T. éste se busca primeramente dentro de la tabla de OT para verificar que había sido registrada dentro del sistema con anterioridad, al no ser localizada entonces a través de la conexión DSN que existe entre el nuevo sistema de cotización y SiRPAG hace la consulta de su tabla compartida OrdenMecano y busca el número de O.T., en el caso de ser un reporte sin número de O.T. pasa directamente al siguiente proceso: Tipo de usuario asignándole un número por default. Todo este análisis se presenta en la figura 3.8.

*Proceso Tipo de usuario:* según el tipo de usuario habilitará o no la opción de registrar más reportes sin cerrar los anteriores que estén asignados a la misma O.T. y sus campos correspondientes para actualizar el reporte seleccionado, si se trata de una actividad cuantitativa permitirá habilitar su campo correspondiente para las cifras.

*Proceso Actualizar estatus del reporte:* a partir de elegir si será nuevo reporte o concluir reporte actual, la información asignada en los campos habilitados será tomada para hacer operaciones sobre la misma y actualizar la tabla REPORTE.

*Flujo de datos Número de O.T.:* es la información del número asignado de la O.T. que va a laborar el operador.

*Flujo de datos Actividad sin O.T.:* cuando es un reporte sin número de O.T. se le asigna un número por default.

*Flujo de datos Busca reporte activo:* representa la información que ayuda a localizar si existe algún reporte registrado ese mismo día y que no se encuentre finalizado.

*Flujo de datos Respuesta sobre el reporte activo:* es la información respuesta con todos los datos del reporte activo existente.

*Flujo de datos Reporte inicial:* representa los datos que constituyen al nuevo reporte que se dará de alta.

*Flujo de datos Cifras y producción concluida:* es la información que se le asignará al reporte antes del cierre.

*Flujo de datos Listado de actividades:* representa la información de todas las actividades que se realizan en la empresa, siendo una la que estará en el reporte.

*Flujo de datos Actualiza actividad:* son los datos procesados y listos para añadirse a la tabla REPORTE.

*Almacén Clave:* representa a la tabla CLAVE que posee las diferentes actividades que hacen los operadores en la compañía.

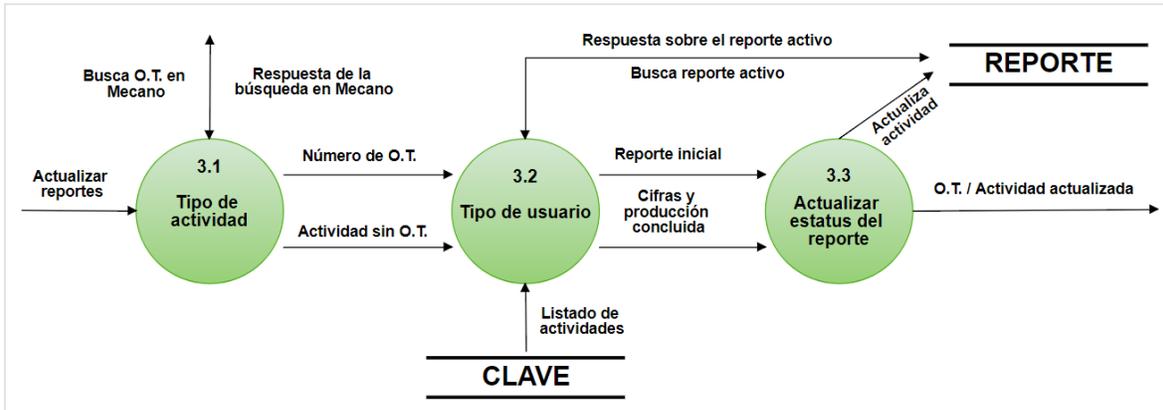


Figura 3.8 DFD Diagrama 1: Registrar reportes.

- **Modelo de máquina de estados**

En este punto describiremos todos y cada uno de los estados por lo que pasa el flujo de información y su comportamiento a partir de los eventos externos que le aplicaremos desde clics en botones de búsqueda, selección de listas, etc. Con el fin de conocer las respuestas posibles y acciones en cada proceso de los diagramas de flujo de datos.

Cada uno de estos diagramas describen el comportamiento de los eventos y como influyen sobre el objeto que pasa por ellos, las transiciones son las flechas que muestran el paso de un estado a otro, estas pueden tener un disparador, un guarda y/o un efecto. El disparador es el evento o señal que causa la transición, el guarda es la condición booleana para que el disparador cause la transición y el efecto es la acción interrumpible consecuente de la transición.

Los modelos de máquinas de estados se presentan en las siguientes páginas comenzando con el modelo de la figura 3.9 que describe el inicio de sesión del usuario, posterior a ella está la figura 3.10 que otorga una vista general del análisis de la bandera para otorgar los permisos correspondientes al usuario.

En la figura 3.11 se muestra un análisis más profundo de la interacción a la hora de actualizar algún reporte de producción, la figura 3.12 describe la creación de una nueva O.T. no registrada previamente en el sistema, la figura 3.13 muestra el registro de una nueva máquina.

La figura 3.14 nos presenta el proceso de consulta de producción, la figura 3.15 el cambio de estados para llegar a la edición o borrado de algún reporte y la figura 3.16 explica los estados para realizar el registro de usuarios.



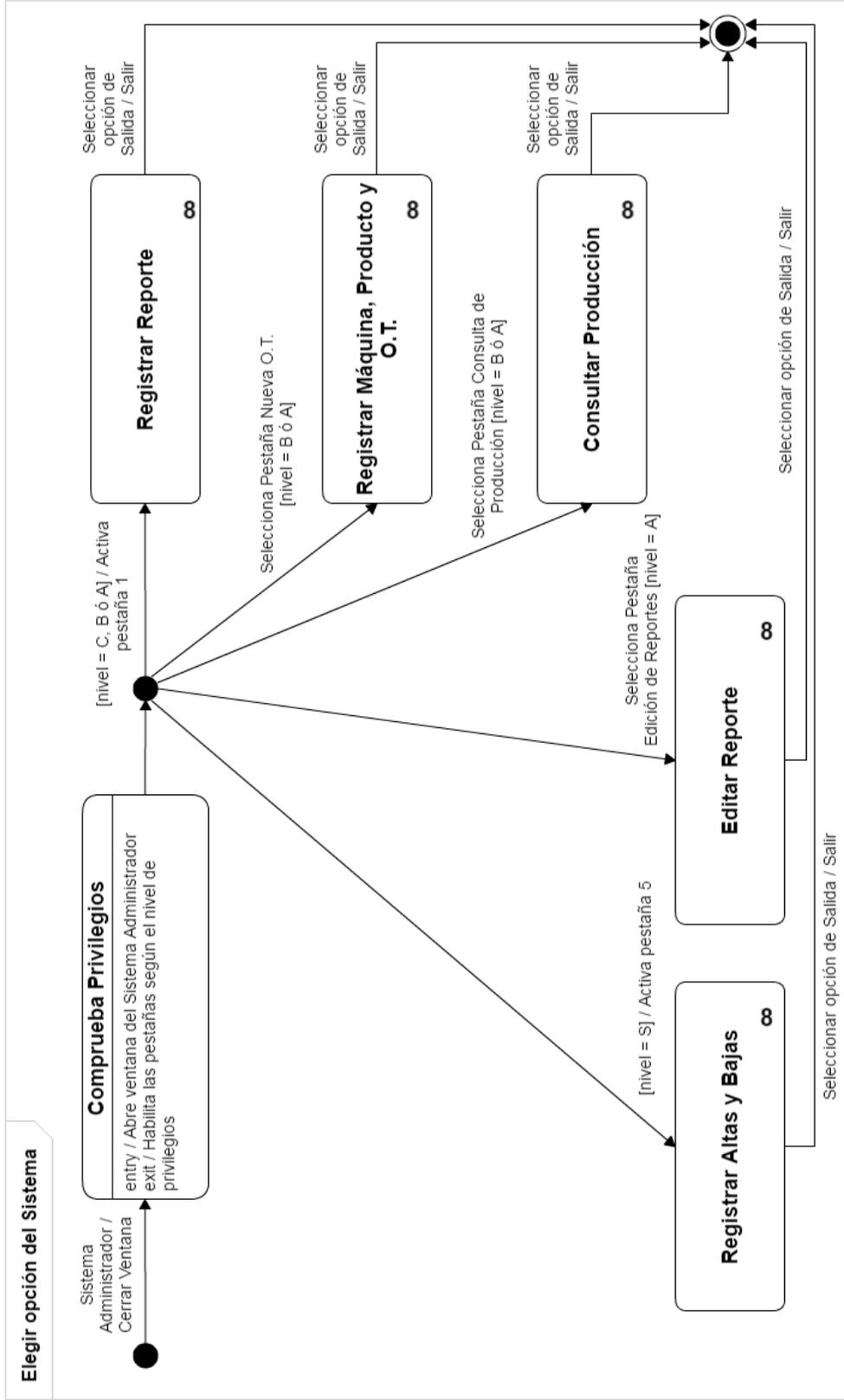


Figura 3.10 DME Diagrama 2: Elegir opción del Sistema.

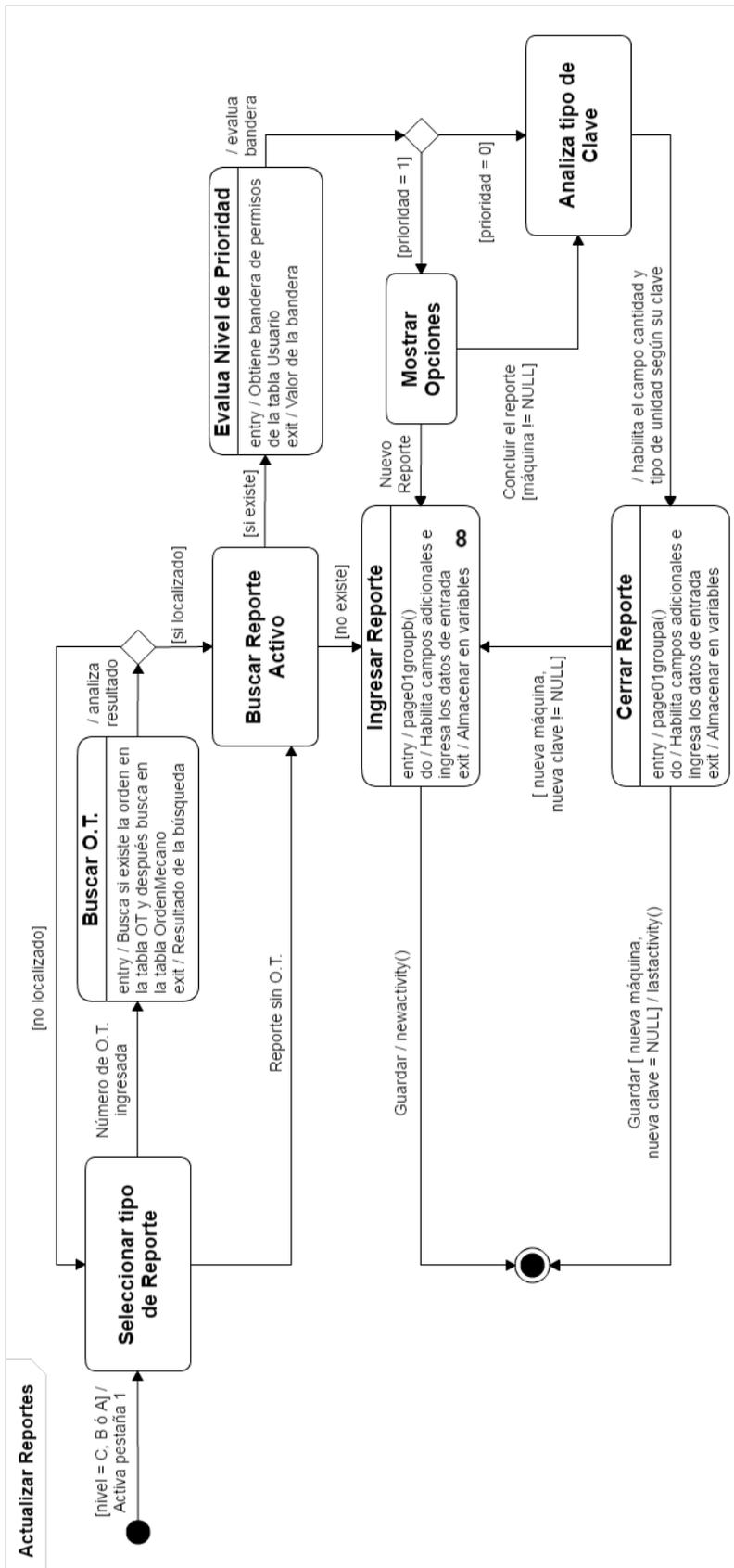


Figura 3.11 DME Diagrama 3: Actualizar Reportes.

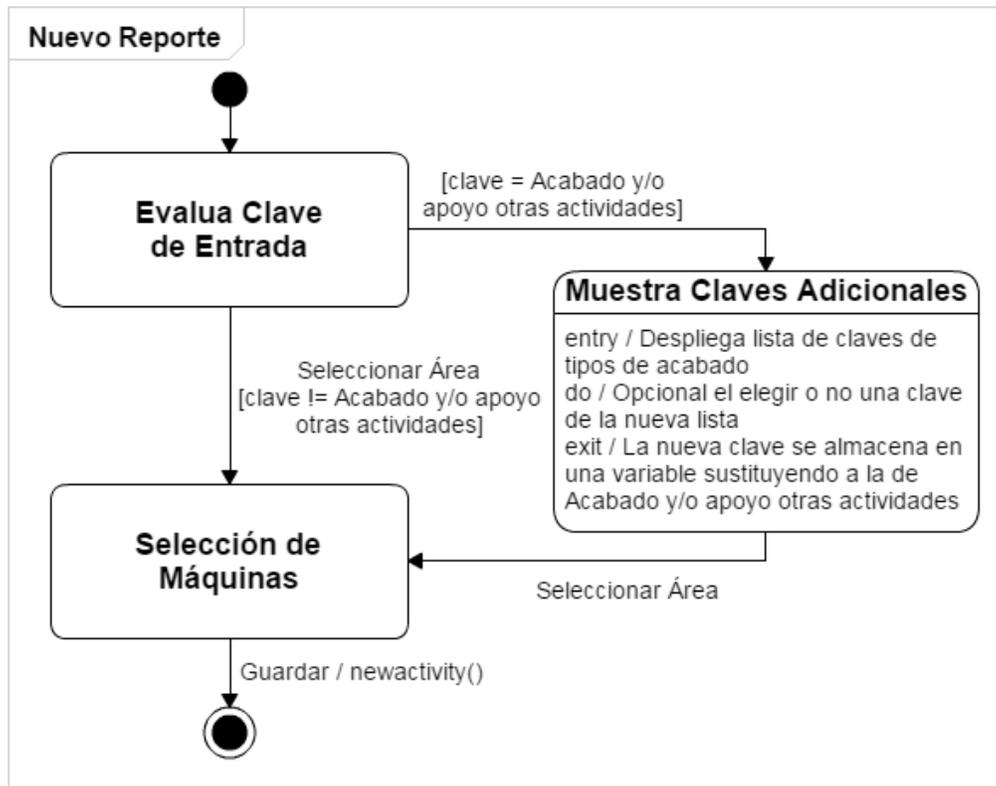


Figura 3.12 DME Diagrama 4: Nuevo Reporte.

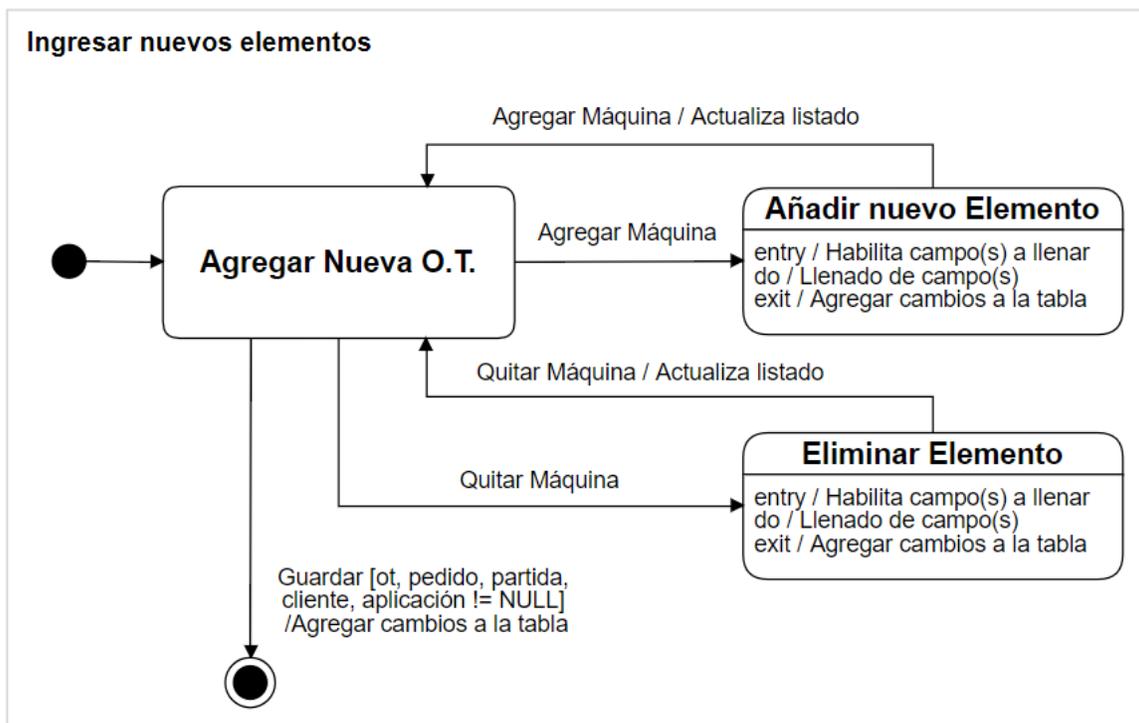


Figura 3.13 DME Diagrama 5: Ingresar nuevos elementos.

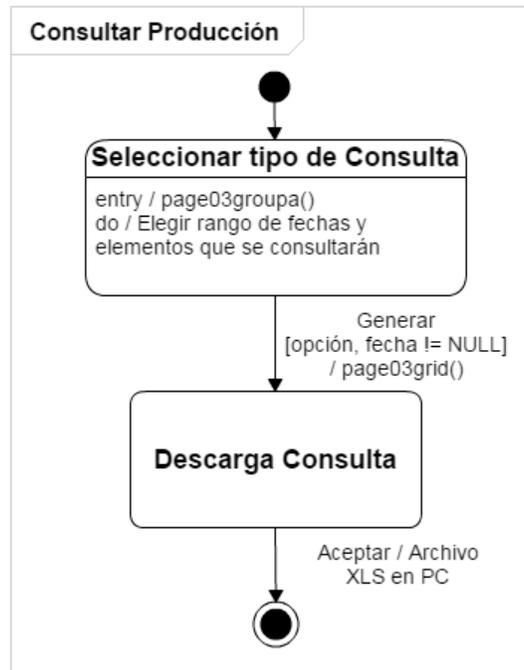


Figura 3.14 DME Diagrama 6: Consultar Producción.

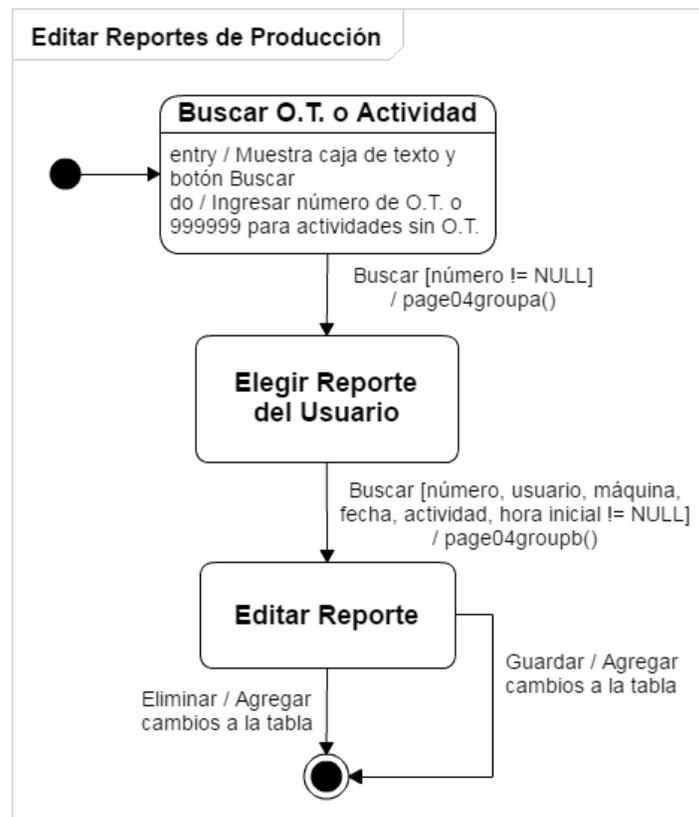


Figura 3.15 DME Diagrama 7: Editar Reportes de Producción.

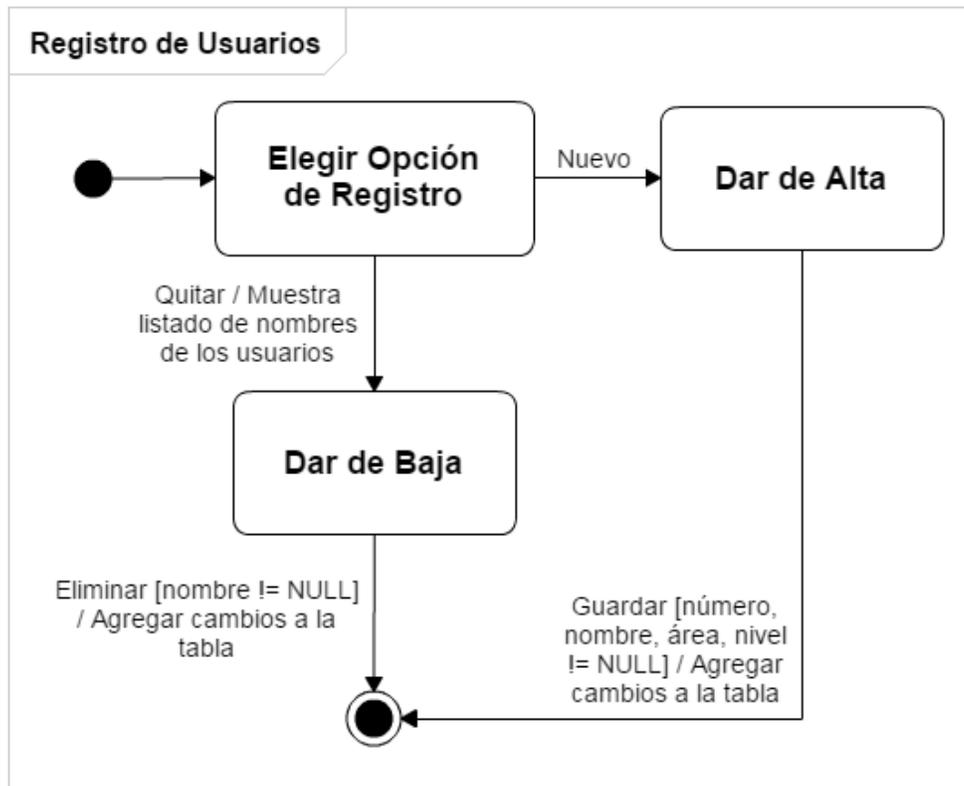


Figura 3.16 DME Diagrama 8: Registro de Usuarios.

### Sesión del Usuario

Se describe cómo se inicia la sesión para poder dar uso al sistema SiRPAG. Tras comprobar que la contraseña y el nombre de usuario sean correctos permite tres opciones: cambiar la contraseña, revisar el historial del usuario o acceder al Sistema Administrador. Si el usuario se equivoca más de 5 veces al ingresar sus datos, el sistema se cerrará como medio de seguridad.

*Estado Limpia Pantalla:* limpia la tabla Grid con la consulta y la oculta después de imprimir o descargar la tabla en un archivo XLS en la PC.

*Estado Entrar al Sistema Administrador:* el símbolo que aparece significa que dentro de éste hay un diagrama aparte que explica a detalle lo que el estado contiene.

*Disparador Seleccionar opción de Salida:* cuando el usuario seleccione alguna opción de salida ya sea el botón de cerrar o alguno de los botones de la barra de acceso rápido la ventana de la consulta se cerrará.

*Efecto Guardar cambios:* se refiere a que en la tabla USUARIO se va a remplazar la contraseña anterior por la nueva escrita por el operador.

## Elegir opción del Sistema

*Estado Comprueba Privilegios:* se habilitan las pestañas que pueden usarse según su nivel de privilegios descritos en la figura 3.4, cada uno de los estados posteriores poseen su propio diagrama de máquina de estado.

*Disparador Seleccionar opción de salida:* es cada una de las opciones de salida que tiene el sistema, la primera es con el botón de cerrar, otros están en la barra de acceso rápido que son el consultar el historial y volver al Inicio de sesión que hacen que automáticamente la ventana de SiRPAG se cierre.

*Efecto Activa Pestaña 1:* quiere decir que al entrar a SiRPAG será la primera pestaña que se verá (según sea el nivel de privilegios).

## Actualizar Reportes

Sólo los usuarios de nivel tipo C, B o A los que pueden usar este apartado, dentro del estado Ingresar Reporte se produce otro diagrama que es el de la figura 3.11.

*Método **page01process()**:* este método es mandado a llamar por el Estado Buscar Reporte Activo, abarcando los estados y correspondientes transiciones del Estado Mostrar opciones y el Estado Analiza tipo de Clave.

*Estado Buscar Reporte Activo:* el método `page01process()` comienza haciendo una consulta en la tabla REPORTE, buscando el número de O.T., nombre del usuario, fecha del día, y sin hora final. Así determina si se requiere concluirla o sólo crear un nuevo reporte.

*Estado Mostrar Opciones:* si el usuario puede dar de alta más de un reporte de la misma O.T. sin concluir el anterior se le permite mostrar las siguientes dos opciones: cerrar algún reporte de los que ya tiene en ejecución o registrar otro. Estos botones sólo se verán hasta llegar a este estado.

*Estado Analiza tipo de Clave:* manda a llamar al método `tipocantidad()` después de corroborar que la actividad seleccionada sea del tipo cuantitativa.

*Estado Cerrar Reporte:* esta acción habilita los campos solicitados en el requerimiento #4.

*Método **tipocantidad()**:* evalúa la bandera que posee la máquina que se registró en el reporte según sea esta y asigna el texto posterior al cuadro de texto donde se colocará la cantidad de producto terminado.

*Método **page01groupa()**:* muestra todos los campos que pertenecen al cerrar un reporte, la actividad que se realizó, la máquina, el cuadro de comentarios con la información escrita si es el caso y la cantidad con su unidad de medida si la actividad lo amerita.

*Método **page01groupb()***: visualiza los campos para un nuevo reporte. La lista de actividades, las áreas de trabajo y máquinas que operan y un cuadro de texto para comentarios.

*Método **newactivity()***: registra en una nueva tupla de la tabla REPORTE el nuevo informe del usuario con los datos seleccionados e ingresados en los campos visualizados por parte del método `page01groupb()`.

*Método **lastactivity()***: localiza el reporte del usuario en la tabla REPORTE y guarda la información adicional para el cierre, a su vez ejecuta el método `totaltime()`.

*Método **totaltime()***: tomando la hora de inicio del reporte de usuario y la hora final, transforma y calcula el total de horas y minutos que le tomó concluir su actividad al usuario.

*Disparador Guardar*: ejecuta los métodos `page01checknum()`, `binnacle()` y `page01clear()`.

*Método **page01checknum()***: este método se encarga de asignar el valor 0 a los campos en el llenado del reporte que esperen recibir algún valor numérico. Esto evita errores al registrar la información en la tabla REPORTE.

*Método **binnacle()***: registra la acción ejecutada que dio lugar a este método en la tabla BITACORA.

*Método **page01clear()***: limpia valores asignados a los campos habilitados y los oculta.

## Nuevo Reporte

Diagrama que describe de forma detallada lo que ocurre en el estado Ingresar Reporte en el momento que el usuario comienza a registrar los datos de entrada.

*Estado Muestra Claves Adicionales*: las claves a las que se refiere son las mencionadas en el requerimiento #4.

*Estado Selección de Máquinas*: después de seleccionar el área a la que pertenece la máquina y posteriormente la máquina.

## Ingresar nuevos elementos

*Estado Agregar Nueva O.T.*: ejecuta el método `page02groupa()` para mostrar todos los campos correspondientes a este apartado. Son 2 opciones las que podemos realizar en este estado, la primera es dar de alta una nueva O.T., la segunda es si se requiere eliminar o añadir una nueva máquina, cada transición que parte de este estado a sus múltiples opciones lo determinarán los disparadores.

*Disparador Guardar:* ejecuta una serie de métodos que son **newactivity()**, **binnacle()** y **page02clear()**. Este último limpia y oculta los campos que había sido habilitados por **page02groupa()**.

*Efecto Agregar cambios a la tabla:* la nueva O.T. se guarda en la tabla OT.

## Consultar Producción

*Método **page03groupa()**:* este método se encarga de habilitar y mostrar los campos que permiten definir la consulta.

*Método **page03grid()**:* después de que el disparador Generar solicita la consulta a la tabla REPORTE según los parámetros y se comprueba que hay información en el query, el método va a asignar las columnas que podrán ser visualizadas en la ventana así como su información correspondiente y lo hará visible al usuario.

*Efecto Archivo XLS en PC:* después de guardar el archivo XLS el método **page03clear()** se ejecutará limpiando la ventana y ocultando lo que mostró el método **page03groupa()**.

## Editar Reportes de Producción

*Disparador Buscar:* este disparador se presenta en dos ocasiones. Esto es correcto, la primera vez buscará si la O.T. existe, y la segunda hará la consulta después de haber sido llenado los campos adicionales para localizar el reporte que busca el usuario.

*Efecto **page04groupa()**:* si existe la O.T. en la tabla REPORTE se habilitarán los listados de: nombre de usuario, máquina, fecha, actividad y hora inicial.

*Efecto **page04groupb()**:* habilita y muestras los elementos que editables del reporte después de haber sido localizado en la tabla.

## Registro de Usuarios

*Estado Dar de Alta:* ejecuta el método **page05groupa()** que habilita y muestras los campos que deben ser llenados para registrar al nuevo usuario.

*Estado Dar de Baja:* habilita y muestra el listado de todos los usuarios registrados y el botón de Eliminar.

▪ **Modelo entidad-relación**

El modelo entidad-relación que se muestra en la figura 3.17 son todos y cada uno de los elementos que componen a las tablas de la base de datos, es necesario recalcar que al hablarse de una tabla no se coloca acentos, por ejemplo, la tabla MAQUINA se registró con ese nombre sin acento, por ello no hago referencia a dicha tabla con acento.

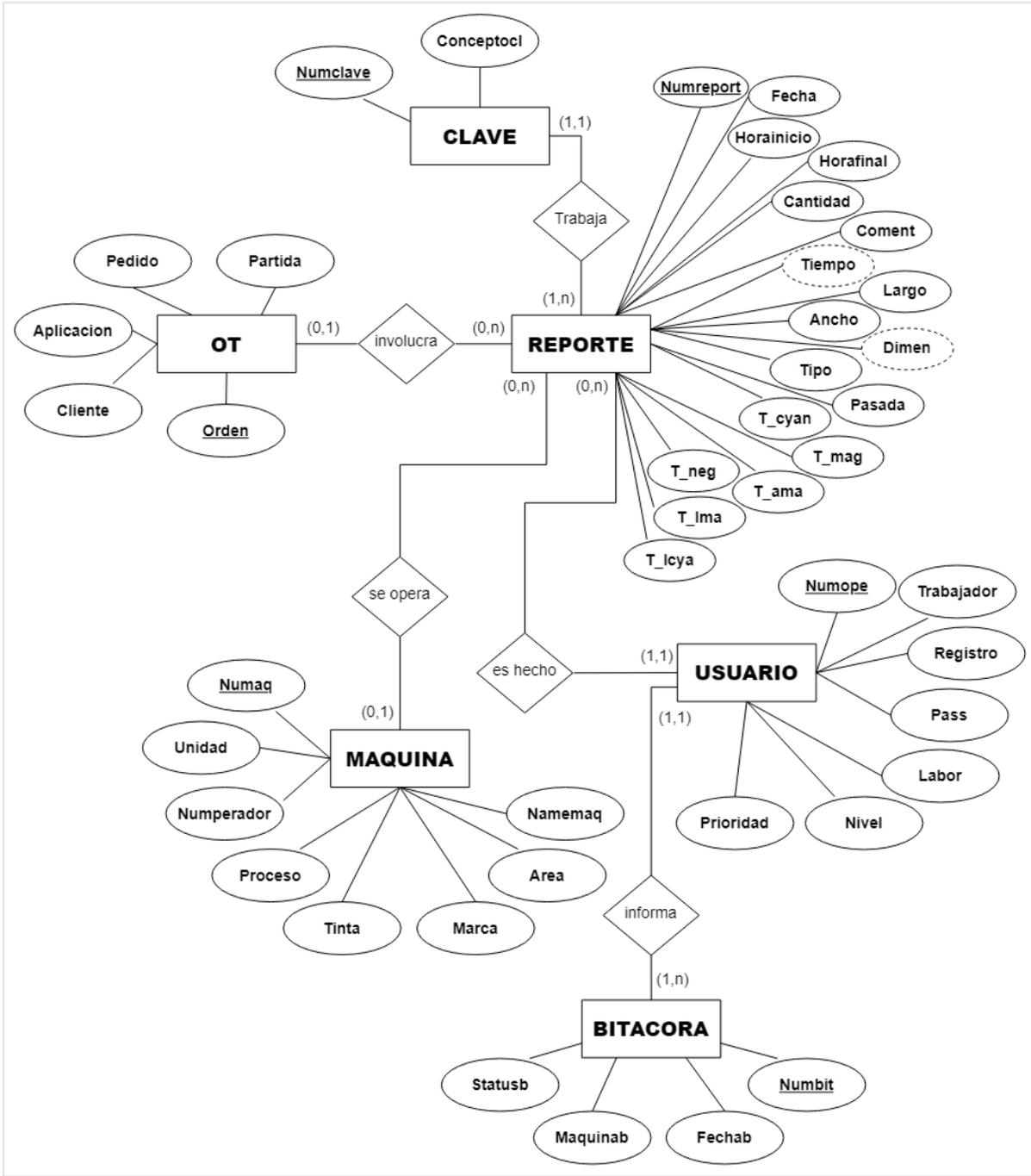


Figura 3.17 Modelo Entidad – Relación de SiRPAG.

- **Diccionario de Datos**

**Tabla OT:** almacena la información que es capturada en el apartado Nueva O.T. y también la que es registrada en Mecano sobre el número de O.T. y pedido al que pertenece, los datos se muestran en la tabla 3.4.

**Tabla 3.4 Tabla OT**

Nombre	Descripción	Tipo	Longitud
<b>Orden (PK)</b>	Número único que corresponde a la O.T. para producción.	Numeric	10
<b>Pedido</b>	Número que corresponde a la solicitud del Cliente conocido como Orden de Compra.	Numeric	6
<b>Partida</b>	Subdivisión del Pedido, con cada partida nueva se crea una O.T.	Numeric	5
<b>Cliente</b>	Nombre del Cliente o Institución.	Character	99
<b>Aplicacion</b>	Producto del Cliente solicitado.	Character	99

**Tabla CLAVE:** almacena todas las actividades que realizan día a día los operadores, los datos se muestran en la tabla 3.5.

**Tabla 3.5 Tabla CLAVE**

Nombre	Descripción	Tipo	Longitud
<b>Numclave (PK)</b>	Número que corresponde a la clave.	Integer (AutoInc)	4
<b>Conceptocl</b>	Actividades que pueden reportar los usuarios.	Character	35

**Tabla MAQUINA:** información de todos los equipos de la empresa que ayudan a la producción de los pedidos que solicitan los clientes, los datos se muestran en la tabla 3.6.

**Tabla 3.6 Tabla MAQUINA**

Nombre	Descripción	Tipo	Longitud
<b>Nummaq (PK)</b>	Número que corresponde a la máquina.	Integer (AutoInc)	4
<b>Namemaq</b>	Nombre de las máquinas de producción.	Character	30
<b>Area</b>	Lugar de la empresa a la que pertenecen.	Character	23
<b>Marca</b>	Identificación comercial del distribuidor.	Character	35
<b>Tinta</b>	La cantidad de tintas que maneja el equipo en caso de que utilice.	Character	5
<b>Proceso</b>	Tipo de trabajo que realiza el equipo.	Character	35
<b>Numperador</b>	Cantidad de operadores que intervienen en el manejo de la máquina e intervención de la producción que genera.	Character	10
<b>Unidad</b>	Bandera que indica que el tipo de cantidad escalar que se utilizará en la producción.	Character	6

**Tabla USUARIO:** conjunto de datos que describen a todos los trabajadores dados de alta, los datos se muestran en la tabla 3.7.

**Tabla 3.7 Tabla USUARIO**

Nombre	Descripción	Tipo	Longitud
<b>Numope (PK)</b>	Número que corresponde al usuario.	Integer (AutoInc)	4
<b>Trabajador</b>	Nombre completo del Usuario registrado.	Character	50
<b>Registro</b>	Número asignado por Recursos Humanos para identificar al trabajador dentro de la nómina.	Numeric	7
<b>Pass</b>	Inicialmente el número de registro del trabajador.	Character	15
<b>Labor</b>	Área dentro de la compañía donde labora.	Character	25
<b>Nivel</b>	Bandera que habilitará los módulos de SiRPAG que podrán ser usados.	Character	2
<b>Prioridad</b>	Bandera que permite dar de alta más de una máquina de la misma O.T. sin tener que dar de baja reportes registrados con anterioridad.	Numeric	2

**Tabla BITACORA:** información considerada relevante sobre los movimientos realizados por los usuarios dentro de SiRPAG, los datos se muestran en la tabla 3.8.

**Tabla 3.8 Tabla BITACORA**

Nombre	Descripción	Tipo	Longitud
<b>Numbit (PK)</b>	Número que corresponde al registro de las acciones del usuario dentro de SiRPAG.	Integer (AutoInc)	4
<b>Numope (FK)</b>	Número que corresponde al usuario.	Numeric	4
<b>Fechab</b>	Fecha y hora de la ejecución de la acción.	DateTime	8
<b>Maquinab</b>	Nombre del equipo de cómputo donde se operaron las acciones.	Character	35
<b>Statusb</b>	Breve descripción de las acciones del usuario.	Character	31

**Tabla REPORTE:** descripción del informe de trabajo del operador, los datos están en la tabla 3.9.

**Tabla 3.9 Tabla REPORTE**

Nombre	Descripción	Tipo	Longitud y decimales
<b>Numreport (PK)</b>	Número único para cada reporte creado dentro del sistema.	Integer (AutoInc)	4
<b>Orden (FK)</b>	Número único que corresponde a la O.T. para producción.	Numeric	10
<b>Numope (FK)</b>	Número que corresponde al usuario.	Numeric	4
<b>Numaq (FK)</b>	Número que corresponde a la máquina.	Numeric	4
<b>Numclave (FK)</b>	Número que corresponde a la clave.	Numeric	4

Nombre	Descripción	Tipo	Longitud y decimales	
<b>Fecha</b>	Fecha del registro del reporte.	Date	8	
<b>Horainicio</b>	Hora de inicio del nuevo reporte registrado.	DateTime	8	
<b>Horafinal</b>	Hora de término del reporte.	DateTime	8	
<b>Cantidad</b>	Total de unidades generadas al concluir el reporte si corresponde a una actividad cuantitativa.	Numeric	10	
<b>Coment</b>	Observaciones relevantes por el usuario para complementar su reporte.	Character	200	
<b>Tiempo</b>	Total de horas y minutos entre la hora inicial y final del reporte.	Character	7	
<b>Largo</b>	Total de metros de largo del material que se usará en los equipos del Área de Rígidos.	Double	8	3
<b>Ancho</b>	Total de metros de ancho del material que se usará en los equipos del Área de Rígidos.	Double	8	3
<b>Dimen</b>	Registro calculado a partir de la multiplicación del largo por el ancho por la cantidad de piezas concluidas en la producción.	Double	8	3
<b>Tipo</b>	Tipo de material con el que se trabajará.	Character	75	
<b>Pasada</b>	Número de veces que se trabajó sobre un área del material dentro de la máquina de Rígidos para ese reporte.	Numeric	2	
<b>T_cyan</b>	Total de litros de tinta cyan cargados a la máquina del Área de Rígidos.	Double	8	3
<b>T_mag</b>	Total de litros de tinta magenta cargados a la máquina del Área de Rígidos.	Double	8	3
<b>T_ama</b>	Total de litros de tinta amarillo cargados a la máquina del Área de Rígidos.	Double	8	3
<b>T_neg</b>	Total de litros de tinta negro cargados a la máquina del Área de Rígidos.	Double	8	3
<b>T_lma</b>	Total de litros de tinta light magenta cargados a la máquina del Área de Rígidos.	Double	8	3
<b>T_lcya</b>	Total de litros de tinta light cyan cargados a la máquina del Área de Rígidos.	Double	8	3

### 3.2.3 Jerarquía de control

Como podemos ver en la figura 3.18 nuestra Rama Principal es la de “Login” o Inicio de Sesión, en este módulo es cuando se levanta la Base de Datos Principal, la cual alimentará los listados con la información de los usuarios registrados para elegir con cual se iniciará la sesión.

De este módulo principal convergen tres, el módulo del Sistema SiRPAG cuyas actividades se enfocan a la administración de los Reportes de Producción y Registro de Usuarios. El segundo es la Consulta del Historial personal y la tercera el cambio de contraseña.

Como se puede acceder entre el módulo (a) y (b) si así lo desea el usuario se coloca en el mismo nivel de jerarquía. Caso contrario con el cambio de contraseña el cual será llamado por el módulo principal o los módulos (a) y (b), nunca podrán ser llamados los módulos (a) y (b) a partir del módulo (c), por ende, este último es el de menor jerarquía.

De módulo (a) vemos que convergen otros 5 módulos los cuales pueden pasar de uno a otro si cumplen con el nivel de permisos suficiente, por lo que poseen la misma jerarquía uno del otro.

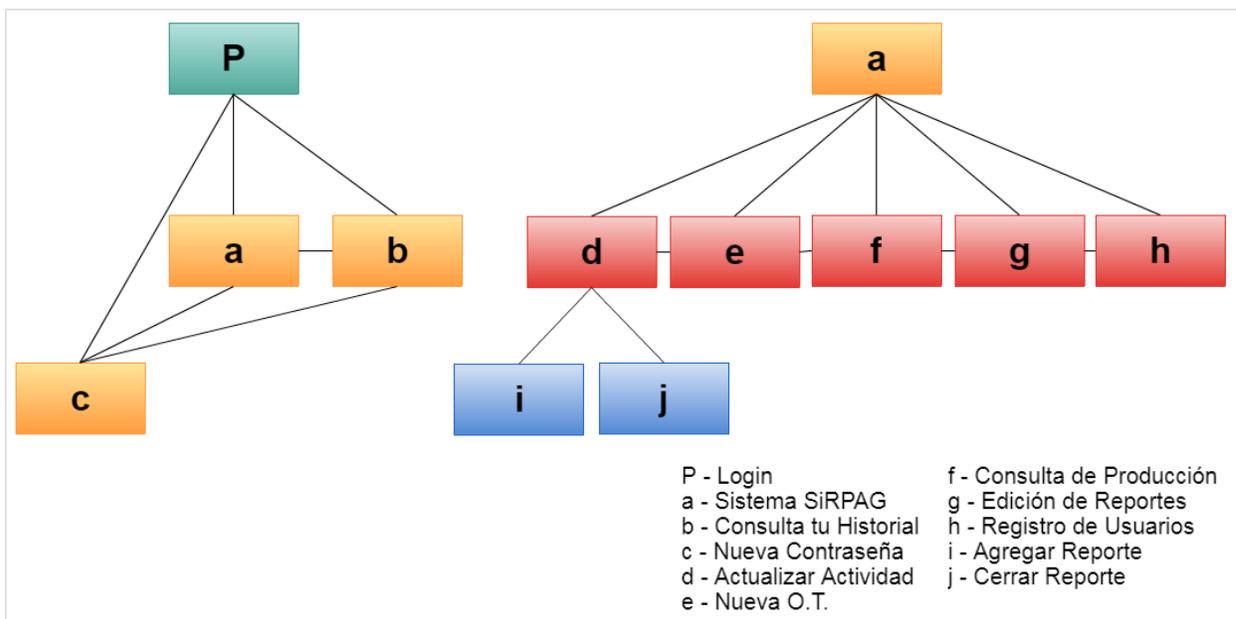


Figura 3.18 Jerarquía de Control

### 3.2.4 Procedimiento del software

Mientras en diagramas anteriores se explica el flujo de los datos y cómo responde el sistema ante eventos internos y externos, en el procedimiento del software explicaremos los métodos y procesos DO CASE involucrados en cada módulo expuesto en la jerarquía de control desde: procedimiento, condición y procesos de repetición, sin entrar a detalle con las decisiones exteriores del usuario que no tenga que ver con el flujo de los datos como la decisión de permanecer en la ventana, cambiar de ventana, cancelar cambios, mensajes de alerta, etc.

- **Descripción del comportamiento a través de diagramas de Flujo**

En el diagrama de Jerarquía de Control se presentan 11 módulos, los cuales se describen a continuación en diagramas de flujo. La figura 3.19 muestra el diagrama principal y sus componentes, la figura 3.20 describe al diagrama del módulo (a), la figura 3.21 al módulo (b), la figura 3.22 al módulo (c) y la figura 3.23 al módulo (d).

Los módulos (i) en la figura 3.24 y (j) localizado en la figura 3.25 son acciones detalladas, por lo que requieren de una mejor explicación, no hay manera de acceder a ellos sino por el módulo (d). Por último, el módulo (e) se localiza en la figura 3.26, el módulo (f) en la figura 3.27, el módulo (g) en la figura 3.28 y el módulo (h) en la figura 3.29.

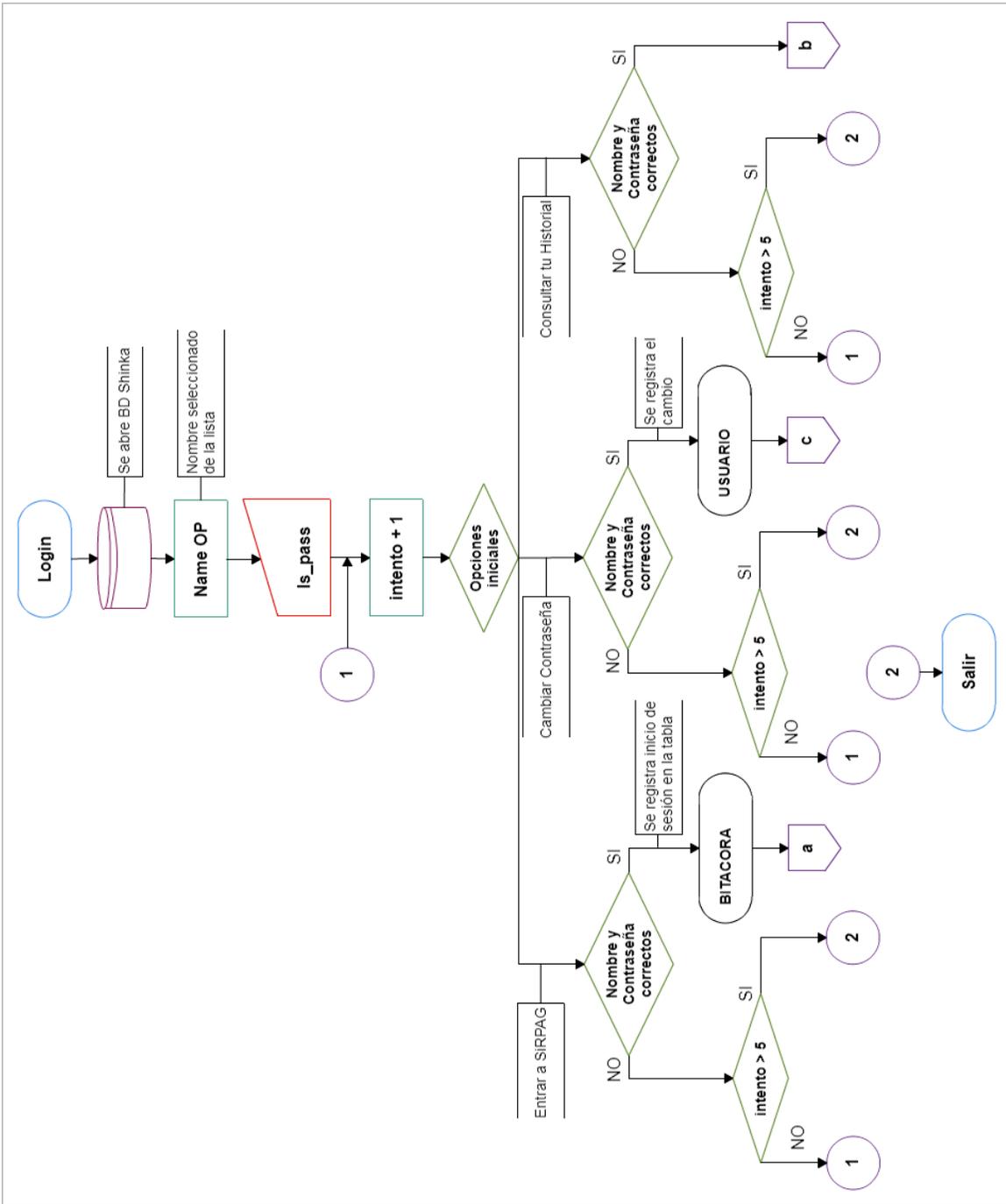


Figura 3.19 Diagrama de Flujo: Módulo Principal

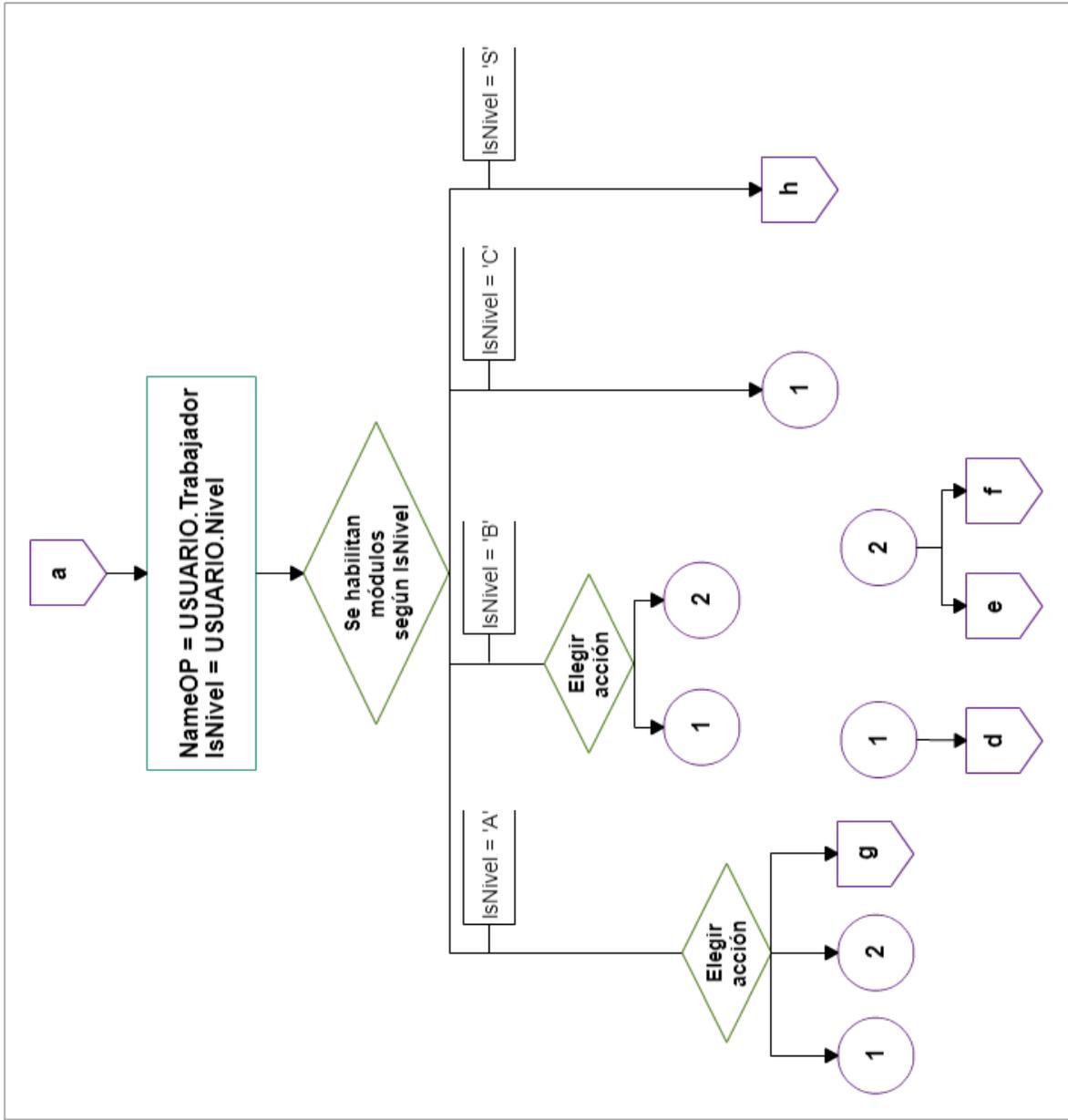


Figura 3.20 Diagrama de Flujo: Módulo A

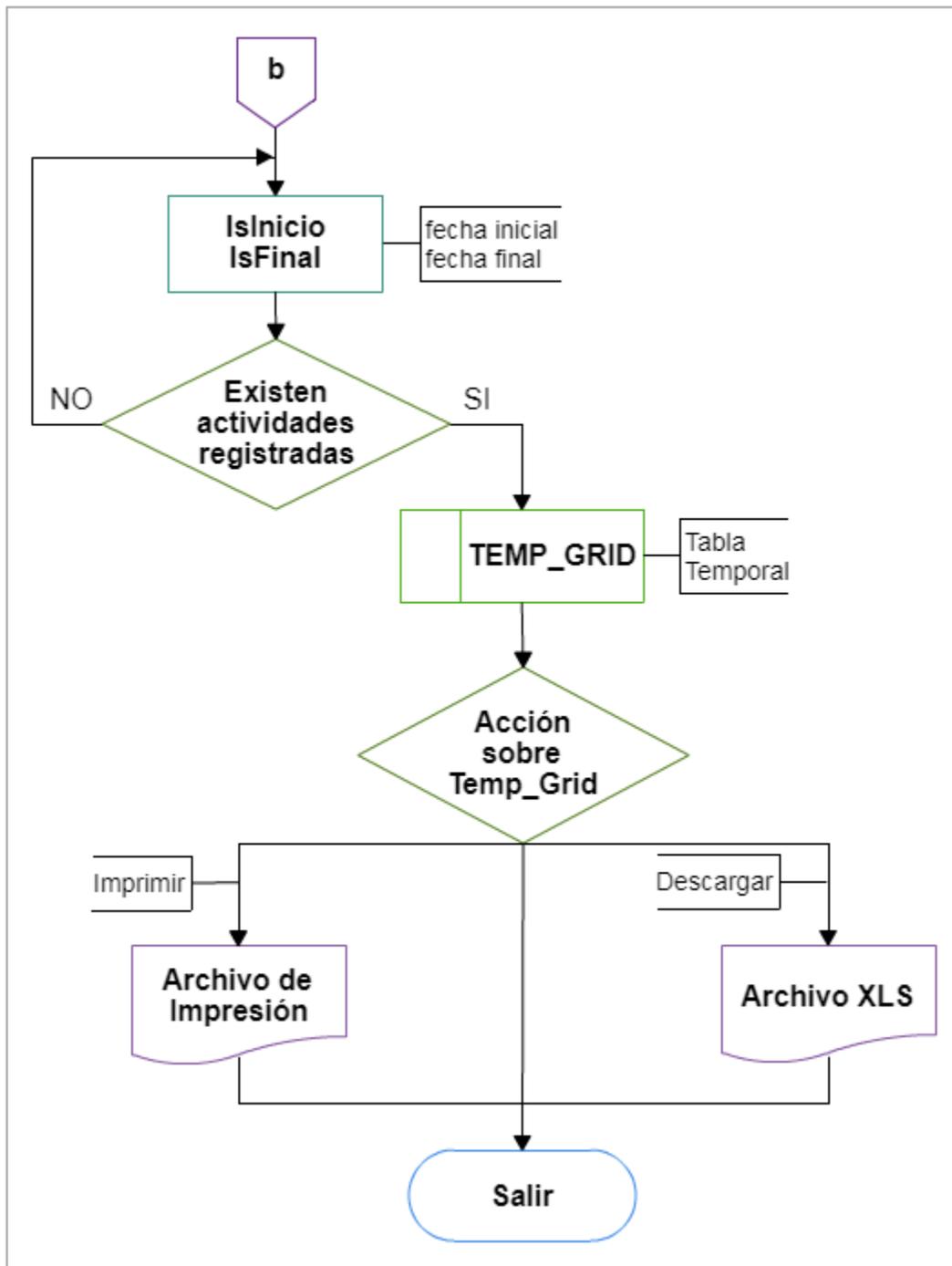


Figura 3.21 Diagrama de Flujo: Módulo B

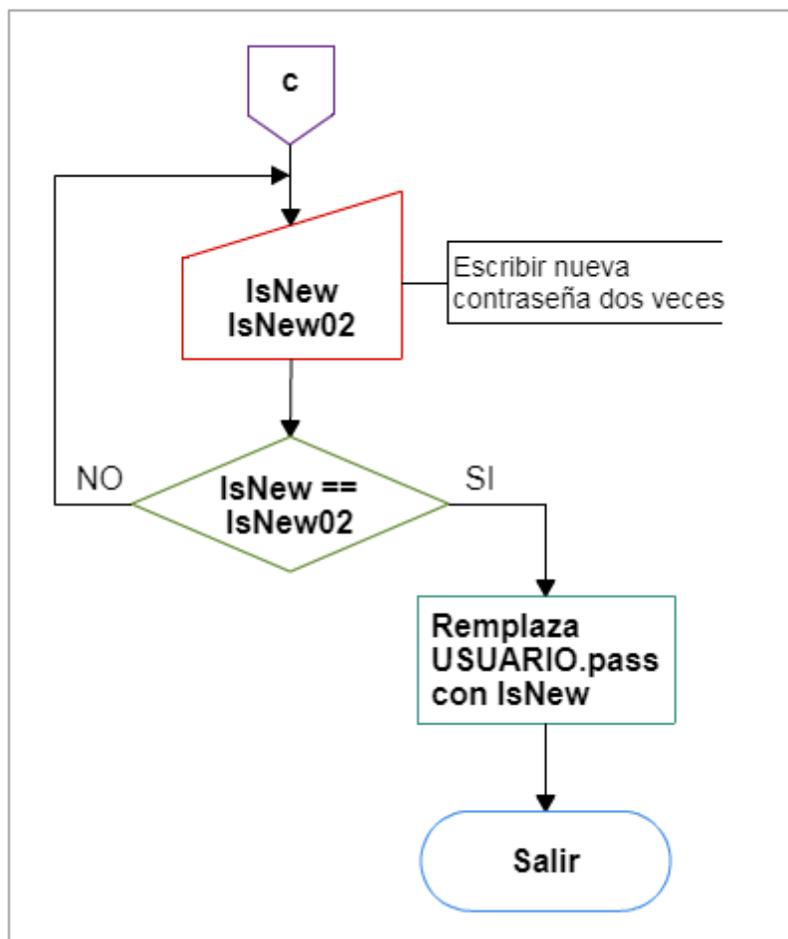


Figura 3.22 Diagrama de Flujo: Módulo C

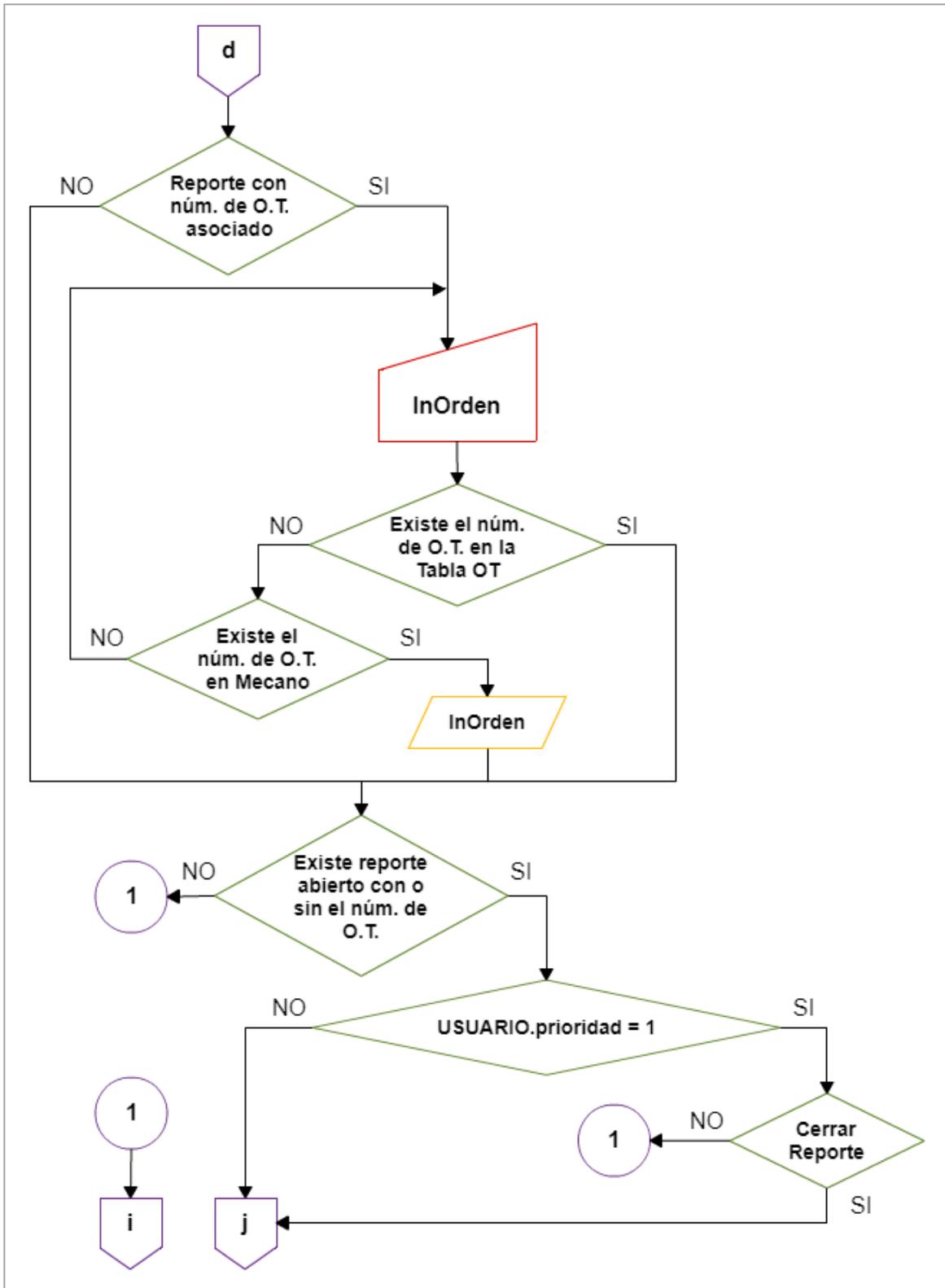


Figura 3.23 Diagrama de Flujo: Módulo D

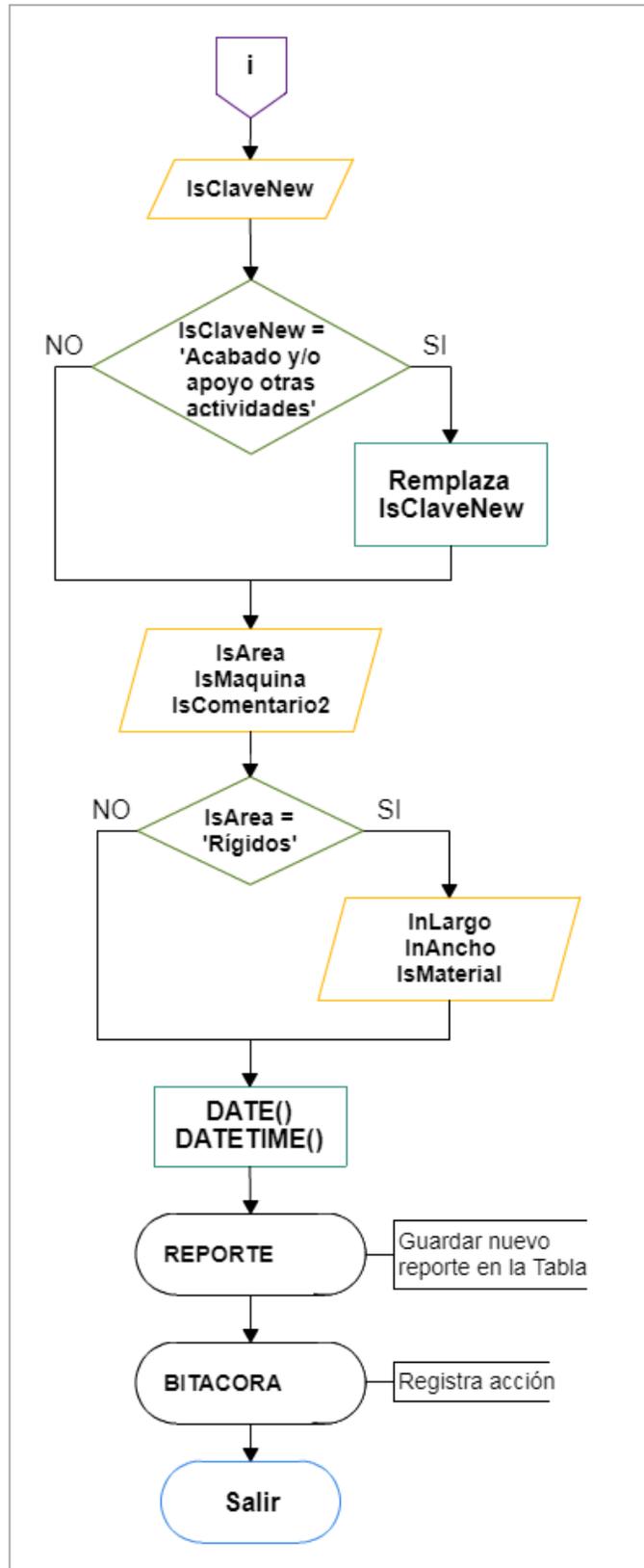


Figura 3.24 Diagrama de Flujo: Módulo I

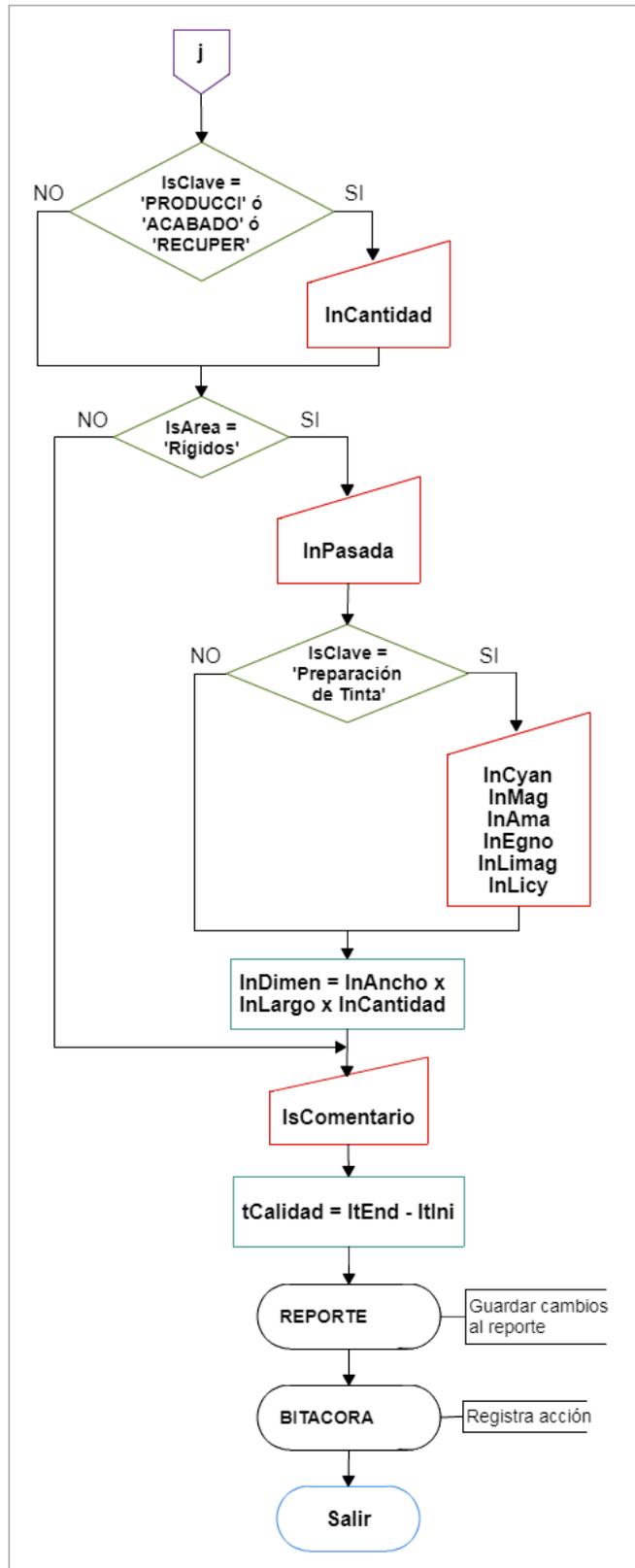


Figura 3.25 Diagrama de Flujo: Módulo J

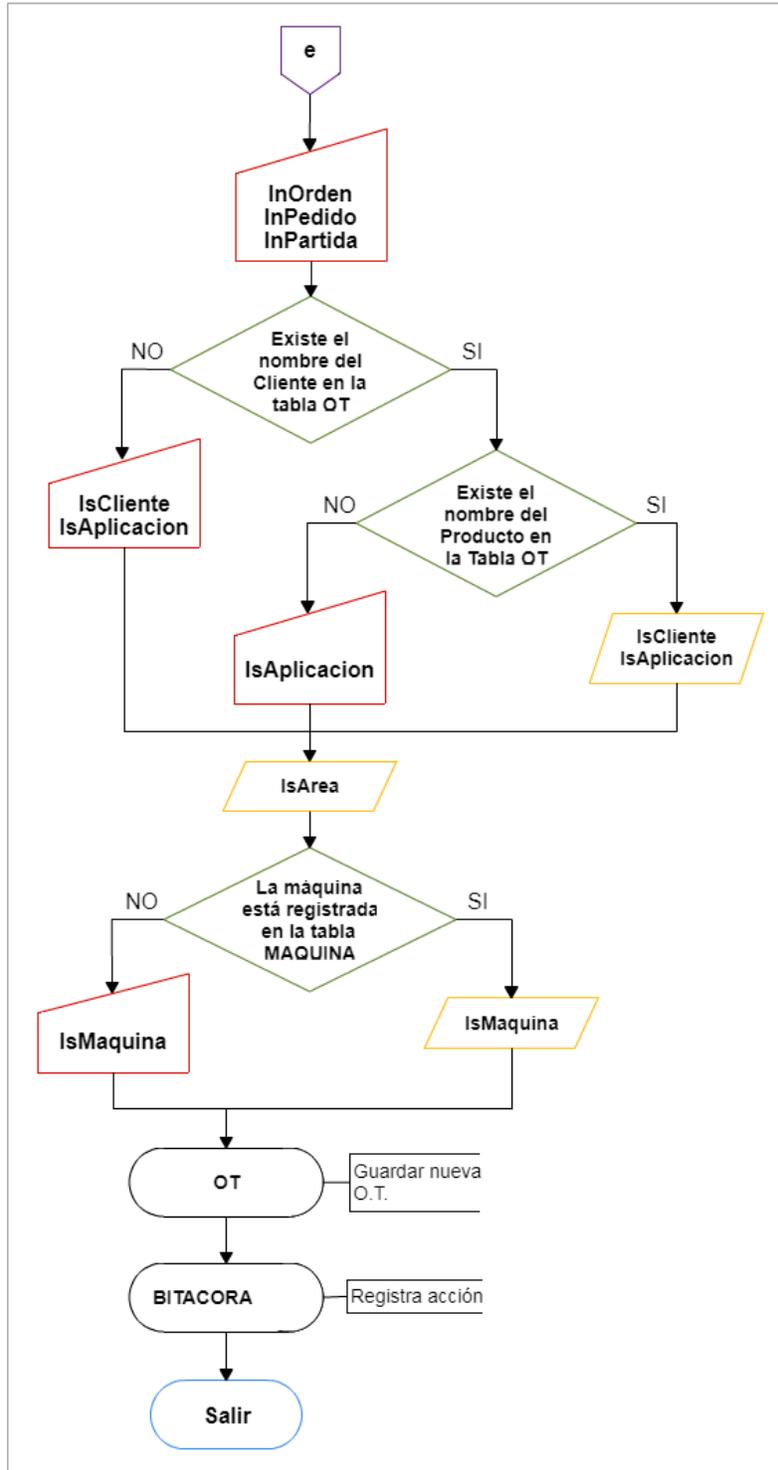


Figura 3.26 Diagrama de Flujo: Módulo E

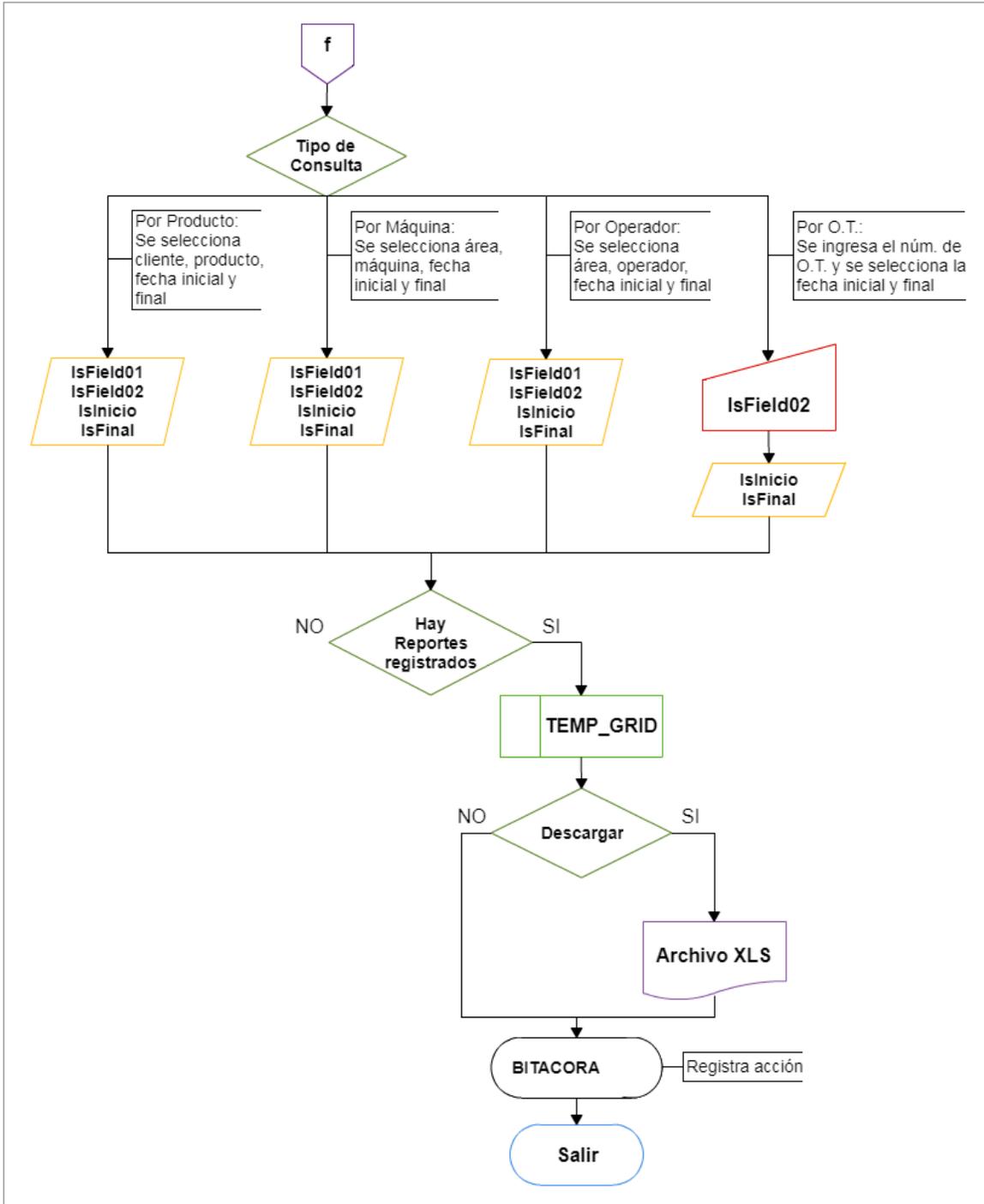


Figura 3.27 Diagrama de Flujo: Módulo F

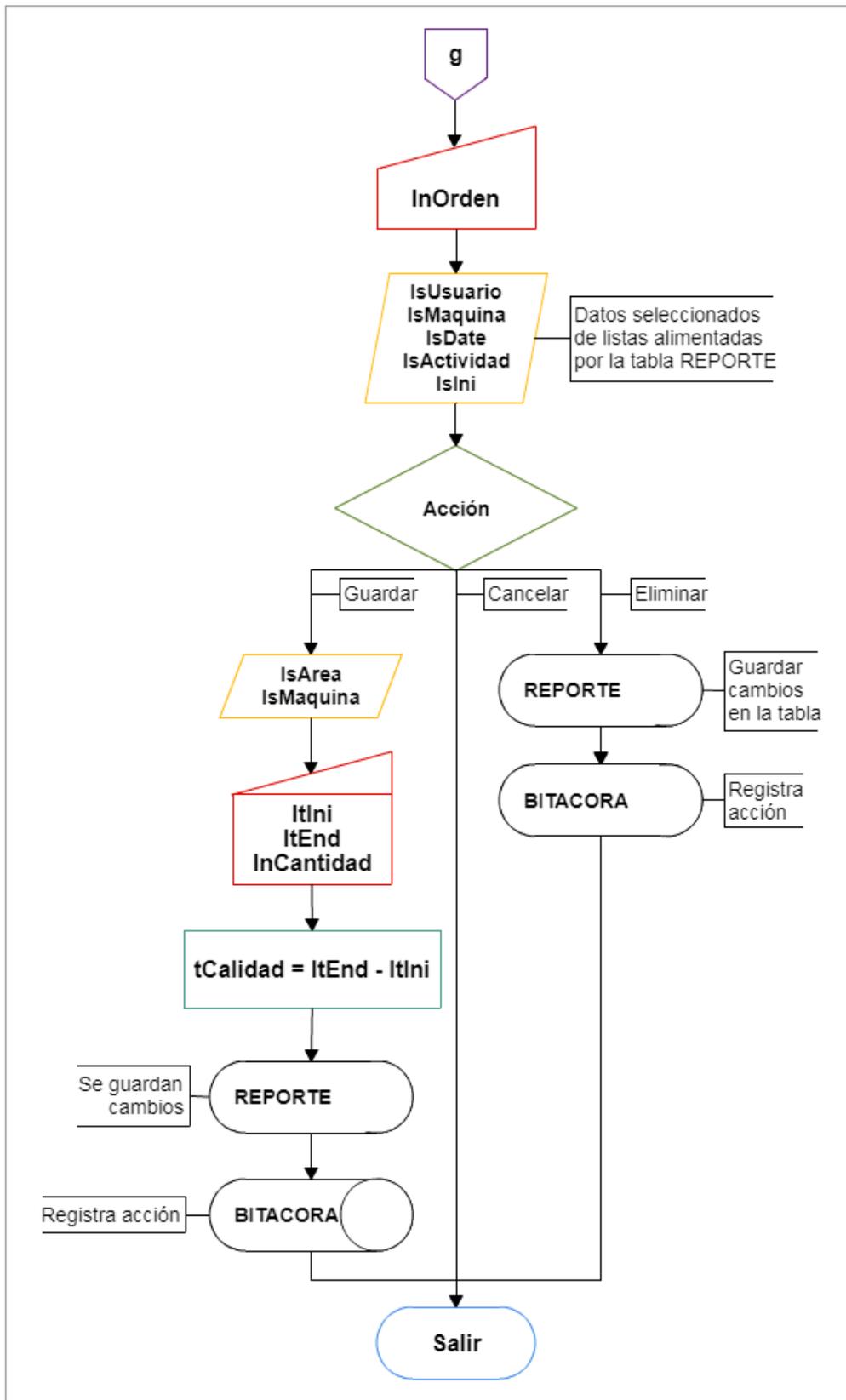


Figura 3.28 Diagrama de Flujo: Módulo G

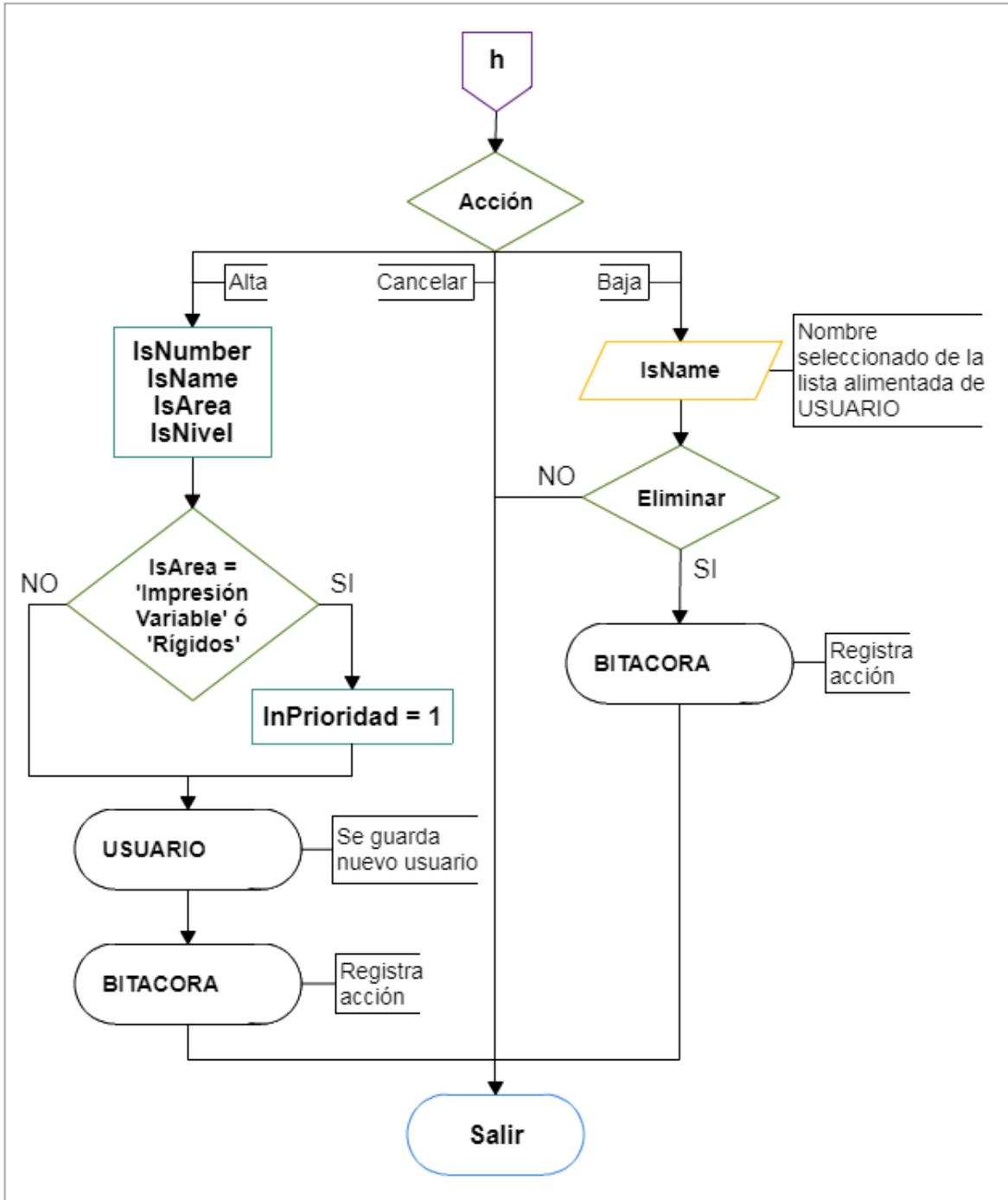


Figura 3.29 Diagrama de Flujo: Módulo H

- **Descripción de los métodos a través de modelado de objetos**

A continuación, en la figura 3.30 se presenta el diagrama de clases heredado, donde la clase más simple hereda sus atributos y métodos a la clase siguiente y de esta manera crea al objeto Form.

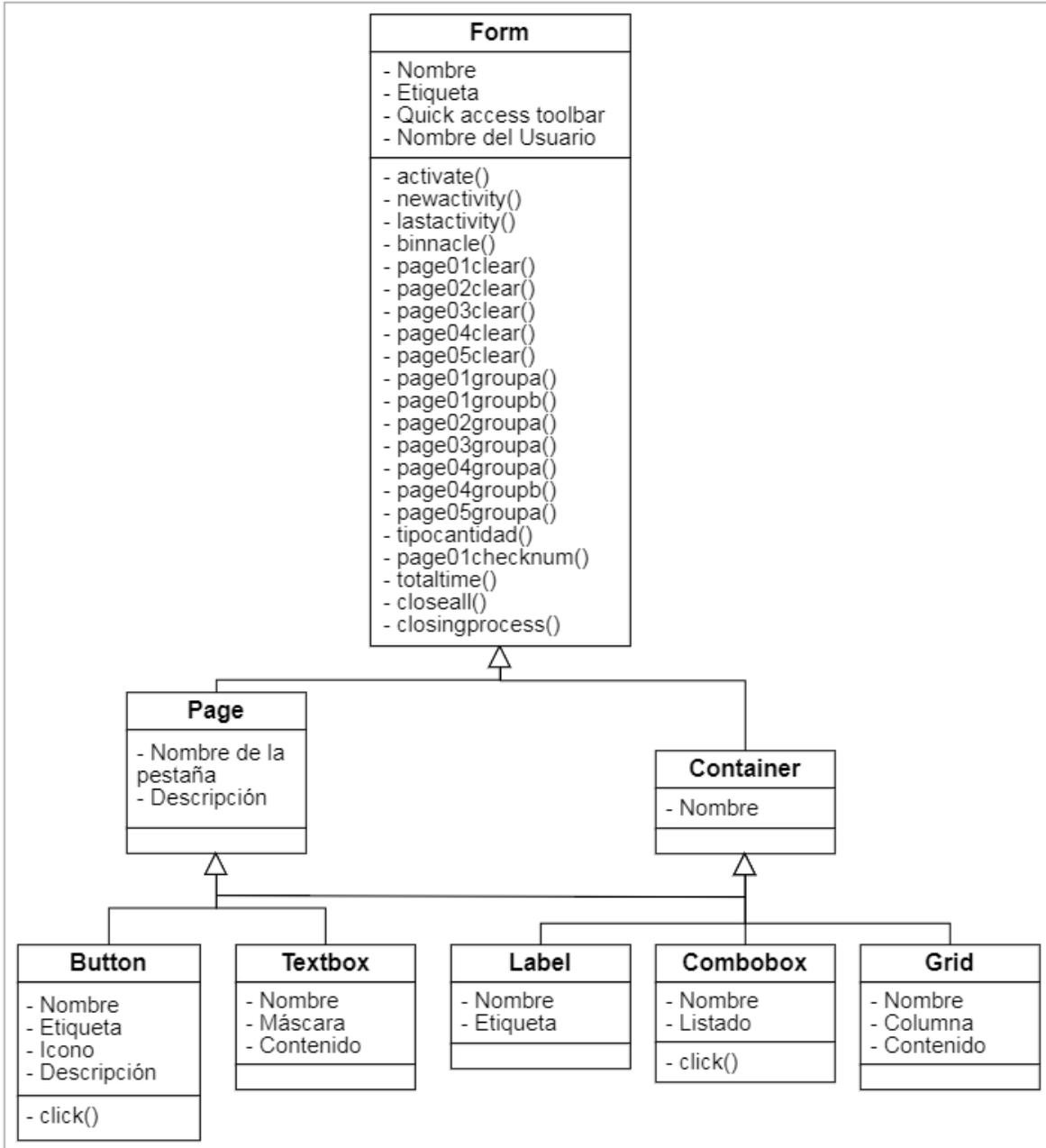


Figura 3.30 Jerarquía de Clases del Formulario

### 3.2.5 Diseño de la interfaz

#### **Diagrama de navegación de interfaz**

El diagrama de la figura 3.31 que representa la interacción Cliente-Interfaz del sistema comienza con la ejecución del módulo de control (MC), el símbolo (p) se entiende que al “pulsar” el botón cuyo nombre se describe después del símbolo se ejecuta y acciona una nueva ventana. Las sentencias OK y ERROR se refieren a si el método o serie de comandos son ejecutados de manera correcta o por lo contrario detectaron algún tipo de error, el mensaje o ventana que se accionará será distinta.

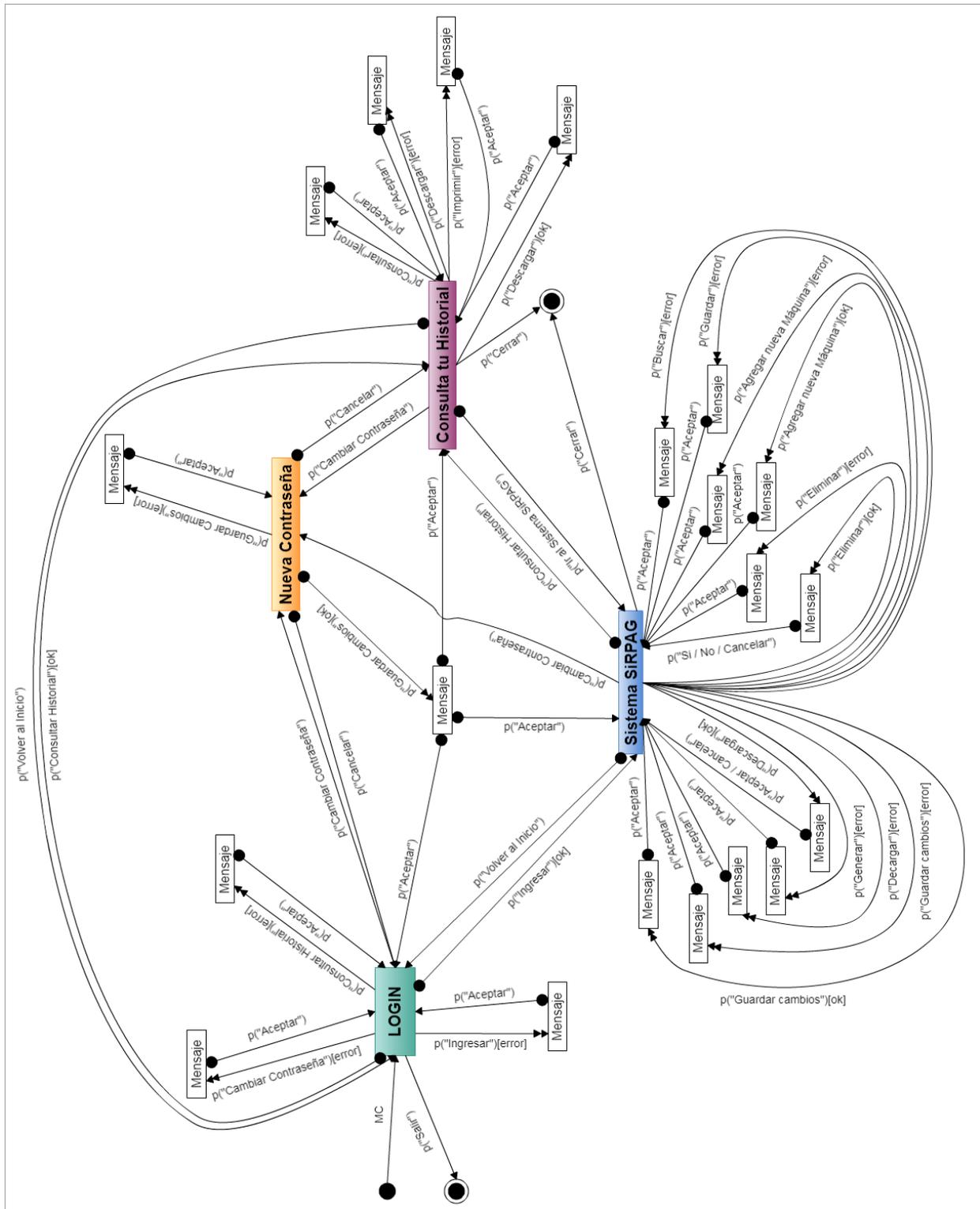


Figura 3.31 Diagrama de Interfaz de Navegación

## Ventas de interfaz de usuario

Se requiere describir cada una de las pantallas a través de Ventanas IU para constatar el cumplimiento de los requerimientos y lo que se describe en el Diagrama de Navegación de la Interfaz.

Adicional a lo que ya se explicó en el DME de la Sesión del Usuario, se puede apreciar en la figura 3.32 y descrita en la tabla 3.10 todos los elementos que componen al inicio de sesión, en el segundo tercio del menú izquierdo aparece un manual operativo que puede ser consultado al pulsar el checkbox correspondiente.

Tabla 3.10 Inicio de sesión

<b>Título:</b> Login		<b>Origen:</b> Inicio de Aplicación / MC
<b>OBJETO</b>	<b>CLASE</b>	
frmLogin	Form	
imgLogin	Image	
themed-ExplorerBar	ExplorerBar	
themedExplorerGroup1	ExplorerGroup	
lblArea	Label	
lblUser	Label	
lblPass	Label	
cbWork	Combobox	
cbUser	Combobox	
txtPass	Textbox	
btnLogin	Commandbutton	
chkChange	Checkbox	
themedExplorerGroup2	ExplorerGroup	
imgChange	Image	
lblChange	Label	
btnRecord	Commandbutton	
themedExplorerGroup3	ExplorerGroup	
imgGuide	Image	
lblGuide	Label	
chkGuide	Checkbox	
btnClose	Themedbutton	

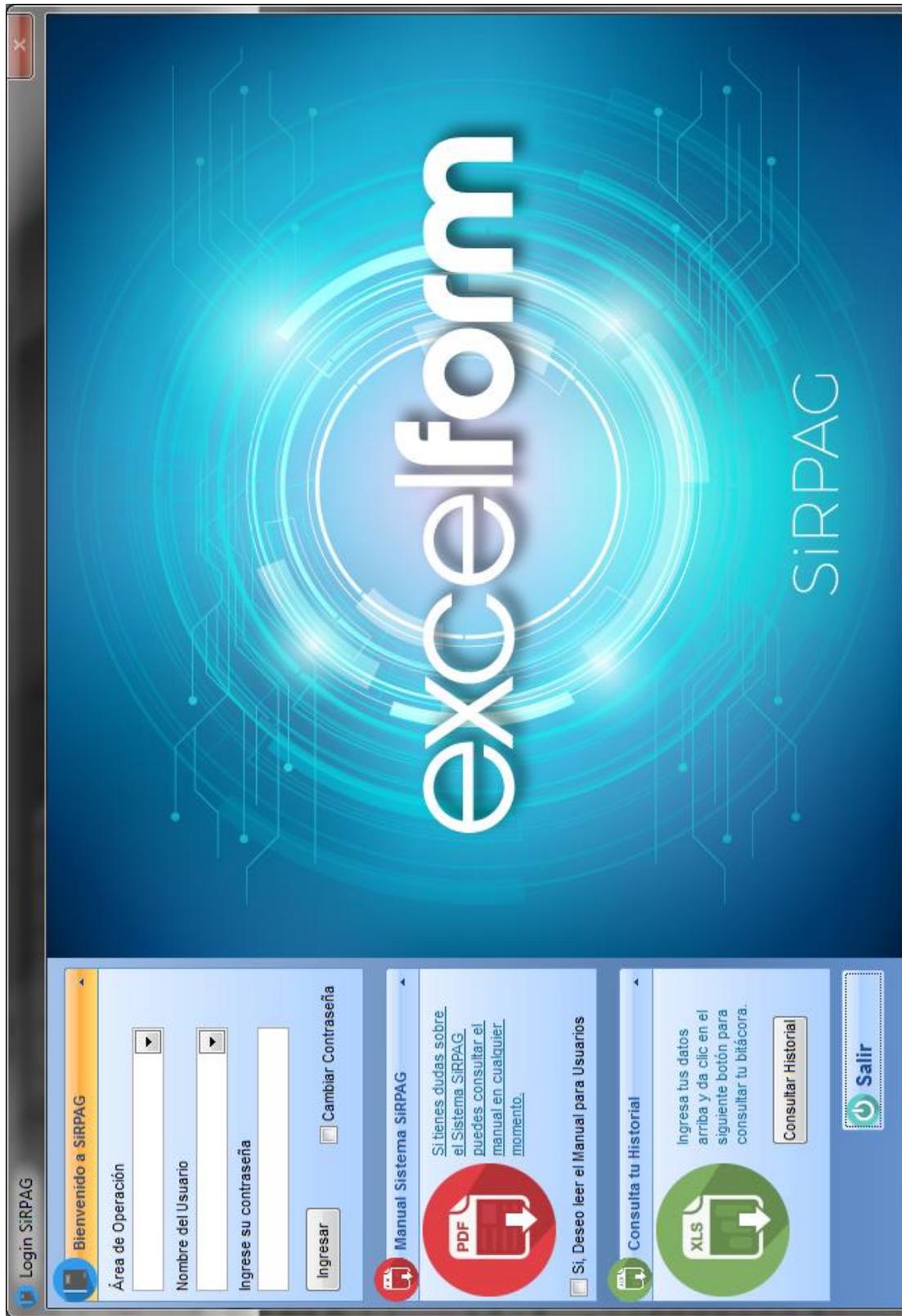


Figura 3.32 Pantalla del Requerimiento #03 Inicio de Sesión

La figura 3.33 muestra la interfaz que verá el usuario al solicitar el cambio de contraseña. Ya no se le pide al usuario la contraseña anterior porque para acceder a esta ventana se requiere ingresarla en el Login SiRPAG. La tabla 3.11 describe cada uno de los elementos que componen a la interfaz.

Tabla 3.11 Nueva Contraseña

<b>Título:</b> Nueva Contraseña <b>Origen:</b> p("Cambiar Contraseña") / Login, Sistema SiRPAG, Consulta tu Historial	
OBJETO	CLASE
frmChange	Form
lbl-Description	Label
lblNewPass	Label
txtNewPass	Textbox
lblCheck	Label
txtCheck	Textbox
btnChange	Commandbutton
btnCancel	Commandbutton

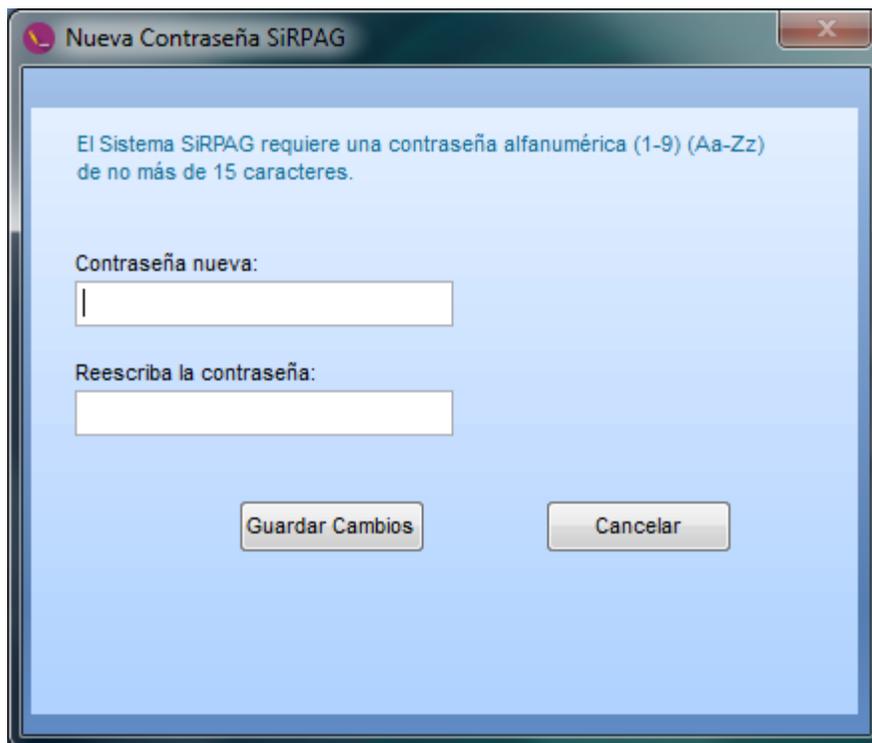


Figura 3.33 Pantalla del Requerimiento #04 Cambio de Contraseña

La figura 3.34 muestra la interfaz que verá el usuario al solicitar la consulta de su historial de trabajo, el objeto gridRecord que vemos en la pantalla no aparece hasta que se realiza una consulta seleccionando las fechas y en el botón Consultar. En el dado caso que no existieran reportes registrados en el rango seleccionado aparece un tipo de mensaje que más adelante veremos cómo aviso. Si existen reportes estos se mostrarán en la grid. La tabla 3.12 describe cada componente de la interfaz.

Tabla 3.12 Consulta tu Historial

<b>Título:</b> Consulta tu Historial		<b>Origen:</b> p("Consultar Historial") / Login, Sistema SiRPAG
<b>OBJETO</b>	<b>CLASE</b>	
frmRecord	Form	
quickaccesstoolbar	Quickaccesstoolbar	
btnSystem	Ribbonbutton	
btnLogin	Ribbonbutton	
btnChange	Ribbonbutton	
btnGuide	Ribbonbutton	
Riboon	Container	
Group01	Group	
lblCaption	Label	
btnSearch	Ribbonbutton	
btnDownload	Ribbonbutton	
btnPrint	Ribbonbutton	
imgComment	Image	
lblComment	Label	
GridRecord	Grid	
lblIni	Label	
lblEnd	Label	
cld_ini	Ctl32_datepicker	
cld_end	Ctl32_datepicker	
lblName	Label	
txtName	Textbox	



Como se puede ver en la figura 3.35 en la esquina inferior derecha hay un dato que es variable y cambia según el usuario, esto para identificar si el reporte que se está registrando pertenece a su cuenta o a la de alguien más. Ésta es la pantalla que verán todos los operadores a la hora de actualizar la información de sus reportes. La tabla 3.13 muestra el detalle de la interfaz.

Tabla 3.13 Elementos generales del Sistema SiRPAG

<b>Título:</b> Sistema SiRPAG Elementos Generales	
<b>Origen:</b> p("Consultar Historial", "Ingresar", "Ir al Sistema") / Login, Consulta tu Historial, Sistema SiRPAG	
<b>OBJETO</b>	<b>CLASE</b>
<b>frmSystem</b>	Form
<b>quickaccesstoolbar</b>	Quickaccesstoolbar
<b>btnRecord</b>	Ribbonbutton
<b>btnLogin</b>	Ribbonbutton
<b>btnChange</b>	Ribbonbutton
<b>btnGuide</b>	Ribbonbutton
<b>containerNameOP</b>	Container
<b>lblName</b>	Label
<b>txtName</b>	Textbox
<b>Riboon</b>	Container
<b>Tabs</b>	Pageframe

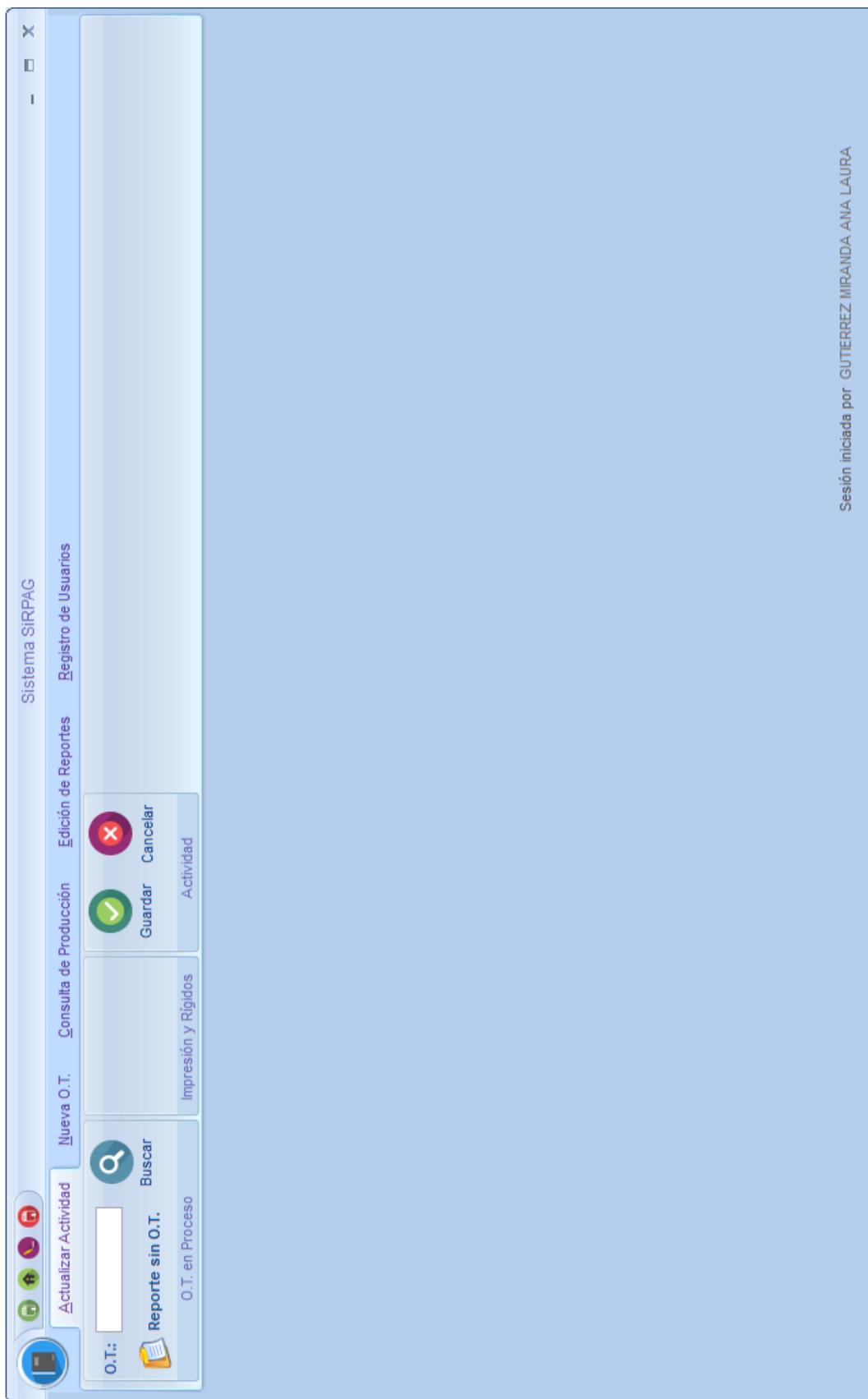


Figura 3.35 Pantalla del Requerimiento #10 Interfaz General de SIRPAG

La interfaz mostrada en la figura 3.36 retrata los elementos que se muestran todos los elementos que aparecerán ya sea para agregar una nueva actividad o actualización. La tabla 3.14 describe el contenido completo que compone a la interfaz.

Tabla 3.14 Primera pestaña del Sistema SiRPAG

<b>Título: Actualizar Actividad</b>	
<b>Origen: p("Ingresar", "Ir al Sistema") / Login, Consulta tu Historial</b>	
<b>OBJETO</b>	<b>LASE</b>
<b>Pagtab01</b>	Page
<b>Group01</b>	Group
<b>lblOt</b>	Label
<b>txtAb</b>	Textbox
<b>txtOt</b>	Textbox
<b>btnSearch</b>	Ribbonbutton
<b>btnActno</b>	Themedbutton
<b>Group02</b>	Group
<b>btnClosema</b>	Ribbonbutton
<b>btnNewma</b>	Ribbonbutton
<b>Group03</b>	Group
<b>btnSave</b>	Ribbonbutton
<b>btnCancel</b>	Ribbonbutton
<b>containerPage01</b>	Container
<b>lblPremach</b>	Label
<b>cbPremach</b>	Combobox
<b>txtPremach</b>	Textbox
<b>lblPreact</b>	Label
<b>txtPreact</b>	Textbox
<b>lblTotal</b>	Label
<b>txtTotal</b>	Textbox
<b>lblBytype</b>	Label
<b>lblComment01</b>	Label
<b>txtComment01</b>	Textbox
<b>Line01</b>	Line
<b>lblNewact</b>	Label
<b>cbNewKey</b>	Combobox
<b>lblActadd</b>	Label
<b>cbActadd</b>	Combobox
<b>lblArea</b>	Label
<b>cbNewarea</b>	Combobox
<b>lblNewmach</b>	Label
<b>cbNewmach</b>	Combobox
<b>lblComment02</b>	Label
<b>txtComment02</b>	Textbox
<b>lblLong</b>	Label

<b>OBJETO</b>	<b>LASE</b>
<b>txtLong</b>	Textbox
<b>lblWdt</b>	Label
<b>txtWdt</b>	Textbox
<b>lblMat</b>	Label
<b>cbMat</b>	Combobox
<b>lblAde</b>	Label
<b>txtAde</b>	Textbox
<b>lblIncy</b>	Label
<b>txtIncy</b>	Textbox
<b>lblInma</b>	Label
<b>txtInma</b>	Textbox
<b>lblInye</b>	Label
<b>txtInye</b>	Textbox
<b>lblInbl</b>	Label
<b>txtInbl</b>	Textbox
<b>lblInlima</b>	Label
<b>txtInlima</b>	Textbox
<b>lblInlicy</b>	Label
<b>txtInlicy</b>	Textbox

Sistema SIRPAG

Registro de Usuarios

Edición de Reportes

Consulta de Producción

Nueva O.T.

Actualizar Actividad

Buscar

O.T. en Proceso

Reporte sin O.T.

Cerrar Reporte Impresión y Rápidos

Nuevo Reporte

Guardar

Cancelar

Actividad

O.T.:

Maquina Utilizada:

Actividad Anterior:

Comentarios:

Cantidad terminada:  Paquete(s)

Nueva Actividad:

Área:

Comentarios:

Acabado en:

Máquina:

Largo:

# de Pasadas:

Ancho:

Cyan:

Negro:

Tipo de Material:

Magenta:

Light Magenta:

Amarillo:

Light Cyan:

Sesión iniciada por GUTIERREZ MIRANDA ANA LAURA

Figura 3.36 Pantalla del Requerimiento #04 Actualizar Actividad

En la figura 3.37 observamos la apariencia del sistema en la sección de agregar una nueva O.T. La tabla 3.15 describe el detalle de cada componente.

Tabla 3.15 Segunda pestaña del Sistema SIRPAG

<b>Título:</b> Nueva O.T.	
<b>Origen:</b> p("Ingresar", "Ir al Sistema") / Login, Consulta tu Historial	
<b>OBJETO</b>	<b>CLASE</b>
Pagtab02	Page
Group01	Group
btnNewot	Ribbonbutton
btnSaveot	Ribbonbutton
btnCancelot	Ribbonbutton
containerPage02	Container
lblNewot	Label
txtAb	Textbox
txtNewot	Textbox
lblNewpe	Label
txtNewpe	Textbox
lblNewpa	Label
txtNewpa	Textbox
lbl-Newcliente	Label
cbNewclient	Combobox
txtNewclient	Textbox
BtnNewclient	Commandbutton
lblNewaplica	Label
cbNewaplica	Combobox
txtNewaplica	Textbox
btnNewaplica	Commandbutton
btnQuitaplica	Commandbutton
lblArea	Label
cbNewarea	Combobox
lblNewmach	Label
cbNewmach	Combobox
btnNewma	Commandbutton
btnQuitma	Commandbutton
txtNewma	Textbox
lblMark	Label
txtMark	Textbox
lblInk	Label
txtInk	Textbox
lblProcess	Label
txtProcess	Textbox
lblTotal	Label
txtTotal	Textbox
lblUnity	Label
cbUnity	Combobox

Sistema SIRPAG

Registro de Usuanos

Edición de Reportes

Consulta de Producción

Nueva O.T.

Actualizar Actividad

Nueva O.T.

Guardar O.T.

Cancelar O.T.

Orden de Trabajo

O.T.:

Pedido:

Partida:

Cliente:

Aplicación:

Área:

Máquina:

Marca:

Tinta:

Proceso:

Operadores:

Unidad de Medida:

Sesión iniciada por GUTIERREZ MIRANDA ANA LAURA

Figura 3.37 Pantalla del Requerimiento #05 Nueva O.T.

En la figura 3.38 observamos la apariencia del sistema en la sección de realizar Consulta de Producción. La tabla 3.16 describe el detalle de cada componente.

Tabla 3.16 Tercera pestaña del Sistema SiRPAG

Título: Consulta de Producción	
Origen: p("Ingresar", "Ir al Sistema") / Login, Consulta tu Historial	
OBJETO	CLASE
Pagtab03	Page
Group01	Group
btnProduct	Ribbonbutton
btnMachine	Ribbonbutton
btnOperator	Ribbonbutton
btnOt	Ribbonbutton
Group02	Group
btnGenerate	Ribbonbutton
btnDownload	Ribbonbutton
containerPage03	Container
lblField01	Label
cbField01	Combobox
lblField02	Label
cbField02	Combobox
lblOt	Label
txtAb	Textbox
txtOt	Textbox
lblIni	Label
cld_ini	Ctl32_datepicker
lblEnd	Label
cld_end	Ctl32_datepicker
grQuery	Grid



En la figura 3.39 se puede ver la interfaz de edición de reportes de producción, la tabla 3.17 describe cada componente existente dentro de esta sección.

Tabla 3.17 Cuarta pestaña del Sistema SiRPAG

<b>Título:</b> Edición de Reportes		<b>Origen:</b> p("Ingresar", "Ir al Sistema") / Login, Consulta tu Historial
<b>OBJETO</b>	<b>CLASE</b>	
Pagtab04	Page	
Group01	Group	
btnSave	Ribbonbutton	
btnRemove	Ribbonbutton	
btnCancel	Ribbonbutton	
ContainerPage04	Container	
lblOt	Label	
txtAb	Textbox	
txtOt	Textbox	
btnSearch	Commandbutton	
Group01	Group	
lblUser	Label	
cbUser	Combobox	
lblMachine	Label	
cbMachine	Combobox	
lblDate	Label	
cbDate	Combobox	
lblClave	Label	
cbClave	Combobox	
lblIni	Label	
cbIni	Combobox	
Group02	Group	
lblPe	Label	
txtPe	Textbox	
lblPa	Label	
txtPa	Textbox	
lblClient	Label	
cbClient	Combobox	
lblAplica	Label	
cbAplica	Combobox	
lblArea	Label	
cbArea	Combobox	
lblMa	Label	
cbMa	Combobox	
lblIni	Label	
txtIni	Textbox	
lblEnd	Label	
txtEnd	Textbox	
lblLot	Label	
txtLot	Textbox	



La última pestaña de SiRPAG la podemos ver en la figura 3.40 donde se registran y se dan de baja los usuarios dentro del sistema. La tabla 3.18 muestra el contenido detallado que compone a la interfaz.

Tabla 3.18 Quinta pestaña del Sistema SiRPAG

<b>Título:</b> Registro de Usuarios	
<b>Origen:</b> p("Ingresar", "Ir al Sistema") / Login, Consulta tu Historial	
<b>OBJETO</b>	<b>CLASE</b>
<b>Pagtab05</b>	Page
<b>Group01</b>	Group
<b>btnNewuser</b>	Ribbonbutton
<b>btnQuituser</b>	Ribbonbutton
<b>Group02</b>	Group
<b>btnSave</b>	Ribbonbutton
<b>btnCancel</b>	Ribbonbutton
<b>imgComment</b>	Image
<b>lblComment</b>	Label
<b>containerPage05</b>	Container
<b>lblPassuser</b>	Label
<b>txtPassuser</b>	Textbox
<b>lblNameuser</b>	Label
<b>txtNameuser</b>	Textbox
<b>cbUser</b>	Combobox
<b>btnQuit</b>	Commandbutton
<b>lblArea</b>	Label
<b>cbArea</b>	Combobox
<b>lblNivel</b>	Label
<b>cbNivel</b>	Combobox
<b>lblAbrev</b>	Label
<b>txtAbrev</b>	Textbox

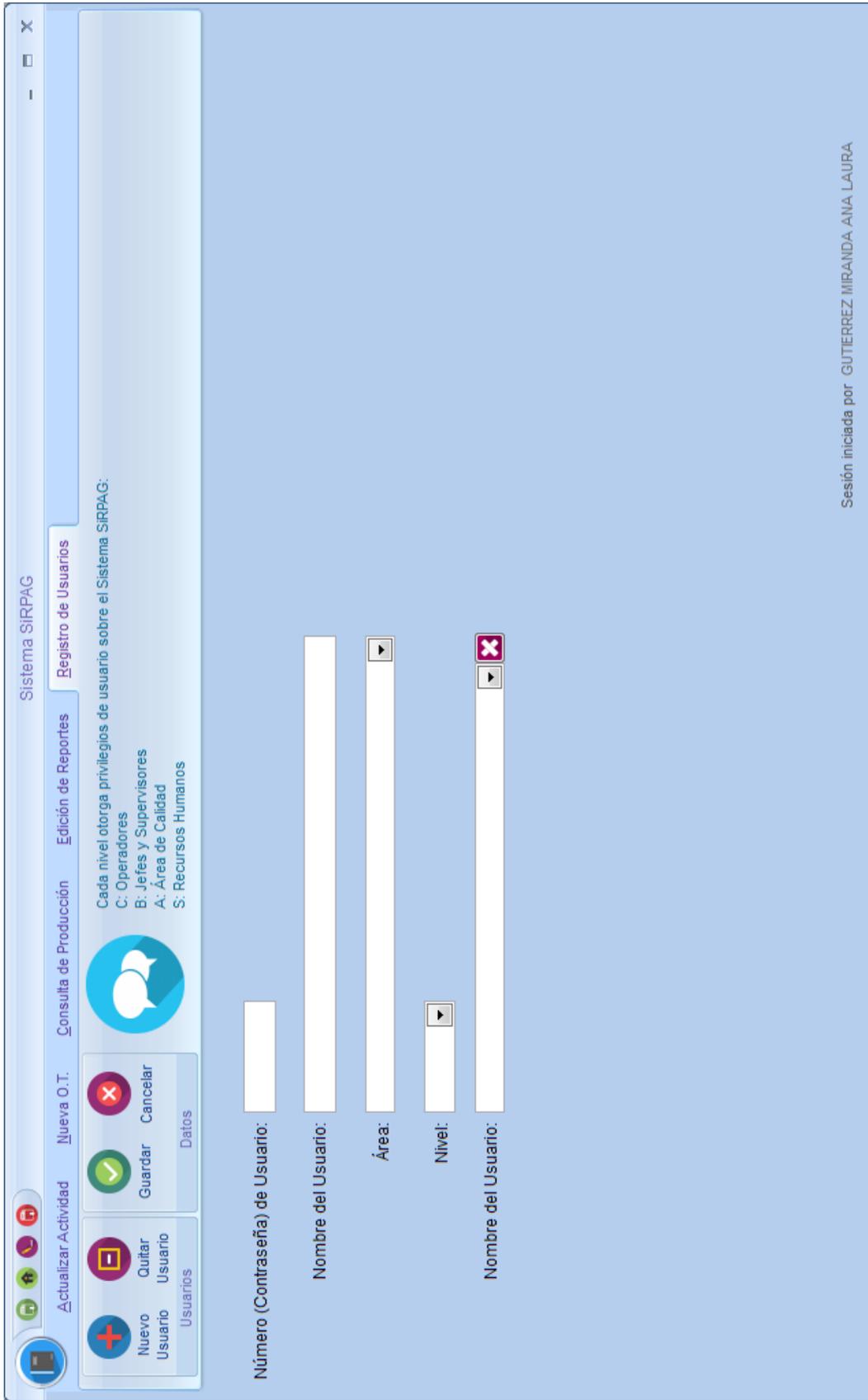


Figura 3.40 Pantalla del Requerimiento #08 Registro de Usuarios

Existen 5 tipos de mensajes presentes a lo largo del manejo de la aplicación con sus títulos y textos correspondientes. El primero que podemos ver en la figura 3.41 es del tipo 3, a partir de la opción seleccionada será el camino que tome el siguiente proceso. El segundo es del tipo 16 que tiene la única opción de aceptar con un icono que representa “alto”, el tercero es el mensaje tipo 0 similar al tipo 16 pero sin el icono.

El cuarto cuadro de diálogo de exploración tipo 16 para asignar la dirección donde se alojará la consulta realizada en formato XLS se muestra en la figura 3.42. Y el quinto un cuadro que muestra las impresoras instaladas, al seleccionar alguna el nombre del equipo se guarda en una variable que a través de una instrucción de programación se imprima la consulta a tamaño carta.

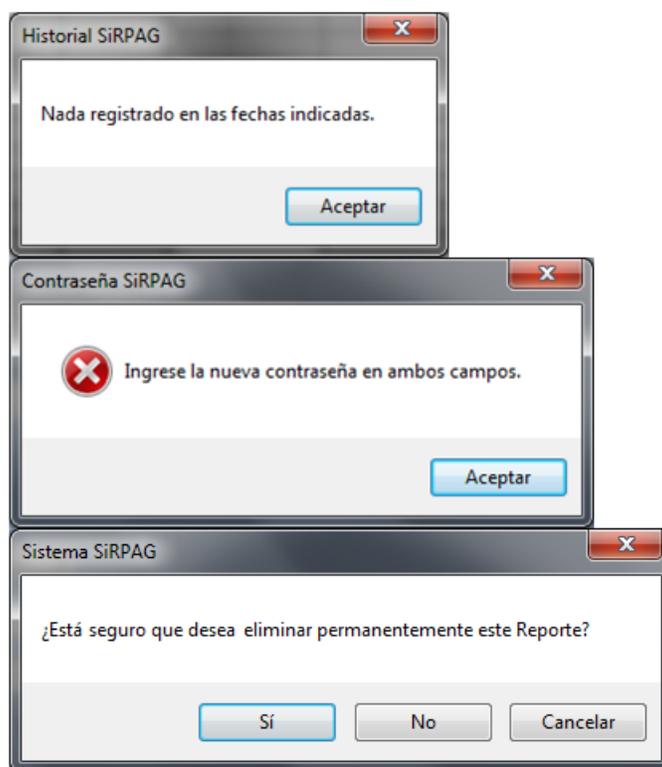


Figura 3.41 Apariencia de los tipos de Mensaje Parte 01.

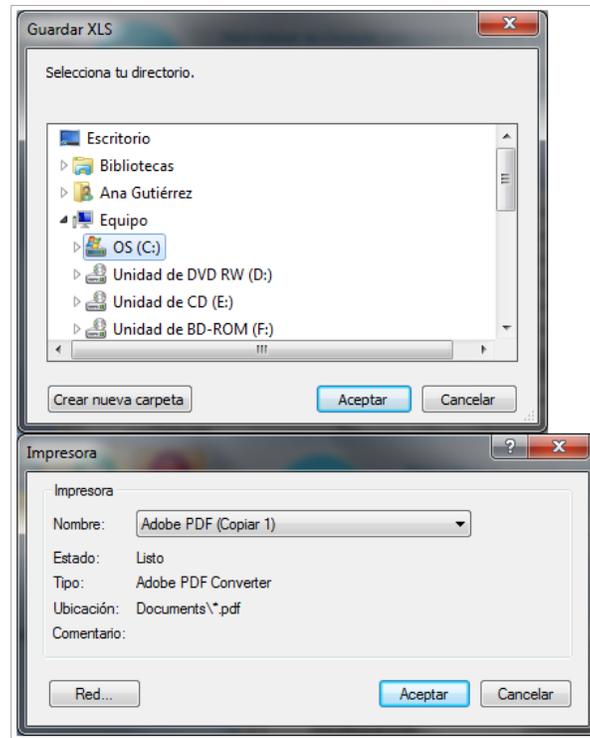


Figura 3.42 Apariencia de los tipos de Mensaje Parte 02.

### 3.2.6 Calidad del diseño

- Validación del diseño

Para la validación del diseño de SiRPAG haremos tres análisis. En los dos primeros se usarán tablas cuya marca X representará que el requerimiento se está cumpliendo en la función señalada. La tabla 3.19 muestra sobre el cumplimiento en asignar los roles y privilegios a la hora de ingresar al sistema.

Tabla 3.19 Roles del usuario

Áreas de Acceso	Operador	Gerentes y Jefes de Área	Área de Calidad	Recursos Humanos
Inicio de Sesión	X	X	X	X
Cambiar Contraseña	X	X	X	X
Consultar Historial	X	X	X	X
Registrar Reportes	X	X	X	
Registrar O.T. o Máquina		X	X	
Consulta de Producción		X	X	
Edición de Reportes			X	
Alta y Baja de Usuarios				X

El segundo análisis en la tabla 3.20 demuestra que se incluyeron los 13 requerimientos específicos mencionados en el apartado 3.1.3 Requerimientos del Desarrollador y en qué momento del proceso se está cumpliendo con el requisito.

Tabla 3.20 Requerimientos **específicos cumplidos**

Proceso en Ejecución	1	2	3	4	5	6	7	8	9	10
Inicio de Sesión		X							X	X
Cambiar Contraseña			X						X	X
Consultar Historial			X						X	X
Registrar Reportes	X			X					X	X
Registrar O.T. o Máquina	X				X				X	X
Consulta de Producción						X			X	X
Edición de Reportes							X		X	X
Alta y Baja de Usuarios								X	X	X

## Usabilidad

Por último, a partir de 10 reglas heurísticas probamos la usabilidad del Sistema comprobando si éste cumple con todos los requisitos para ser accesible, eficaz e intuitivo.

1. Visibilidad del estado del sistema.
  - Todas las ventanas tendrán de cabecera un título que describirá la función que posee.
  - Para la ventana “Consulta tu Historial” y “Sistema SiRPAG” en la esquina inferior derecha estará el nombre del usuario quien inició esa sesión.
2. Utilizar el lenguaje de los usuarios.
  - Botones con descripción clara sin abreviaturas de la acción que realizan.
  - Al posicionar el puntero sobre algún botón se muestra un pequeño cuadro con una breve descripción de lo que realizan.
  - Iconos sobre los botones para interpretar y ubicar rápidamente el botón que se desea.
3. Control y libertad para el usuario.
  - Mensajes de alerta si el usuario comete algún error evitando que un cambio no deseado ocurra.
  - Existe la opción de Cancelar en todas las pestañas de la ventana “Sistema SiRPAG” para limpiar la pantalla de cualquier valor no deseado.
  - Salir del sistema en un paso.
4. Consistencia y estándares.
  - Todos los botones mantienen una nomenclatura estándar: Guardar, Cancelar y Buscar.
5. Prevención de errores.
  - Con la ayuda de los mensajes de alerta y DO CASE evita guardar datos erróneos dentro de las bases de datos.

6. Reconocer mejor que memorizar.
  - Para que el usuario empiece con sus actividades dentro del sistema a partir de lo que requiera se irá demostrando u ocultando elementos según sea el caso para no saturar y evitar la contaminación visual y confundir al operador.
7. Flexibilidad y eficiencia de uso.
  - Para ayudar al usuario y no tener que esperar a soporte técnico se les da la opción de consultar el manual digital disponible en la barra de herramientas superior en la ventana de “Consulta tu Historial”, “Sistema SiRPAG” y como opción disponible en el “Login”.
8. Los diálogos estéticos y diseño minimalista
  - Como se describe en el punto 6, conforme avanza el usuario en los procesos se va limpiando la pantalla de manera dinámica.
  - La fuente y tamaño de la letra se mantienen constantes.
9. Ayuda a los usuarios a reconocer, diagnosticar y recuperarse de los errores.
  - Los mensajes de error y alerta contienen información para diagnosticar y tomar las medidas correspondientes.
10. Ayuda y documentación
  - Descrito en el punto 7, el manual digital se encuentra disponible en diferentes ventanas del Sistema.

### 3.3 Implementación

Esta fase consta en la instalación del nuevo sistema de software con sus correspondientes pruebas y se capacita a los usuarios finales.

- **Método de instalación**

El método de Instalación para este sistema es **piloto y paralelo**. La primera nos indica que sólo una parte de la organización utilizará el sistema en paralelo con el llenado de reportes de forma escrita como hasta ahora se había estado trabajando. El área de la organización que utilizará el sistema piloto será el área de Correo Directo.

Esta área consta de un total de 20 empleados que se dedican a la impresión y el empaque de estados de cuenta bancarios y demás productos de dato variable. Contarán con un total de 7 equipos de cómputo para realizar el llenado de los reportes en SiRPAG. La instalación se realiza cumpliendo con los puntos que aparecen en el Requerimiento #9.

- *Conectarnos a una unidad de red:* nos conectaremos donde se alojará el sistema SiRPAG junto con su base de datos final en una capeta reservada dentro del servidor. Dando clic en la opción Conectar a unidad de red como se ve en la figura 3.43.

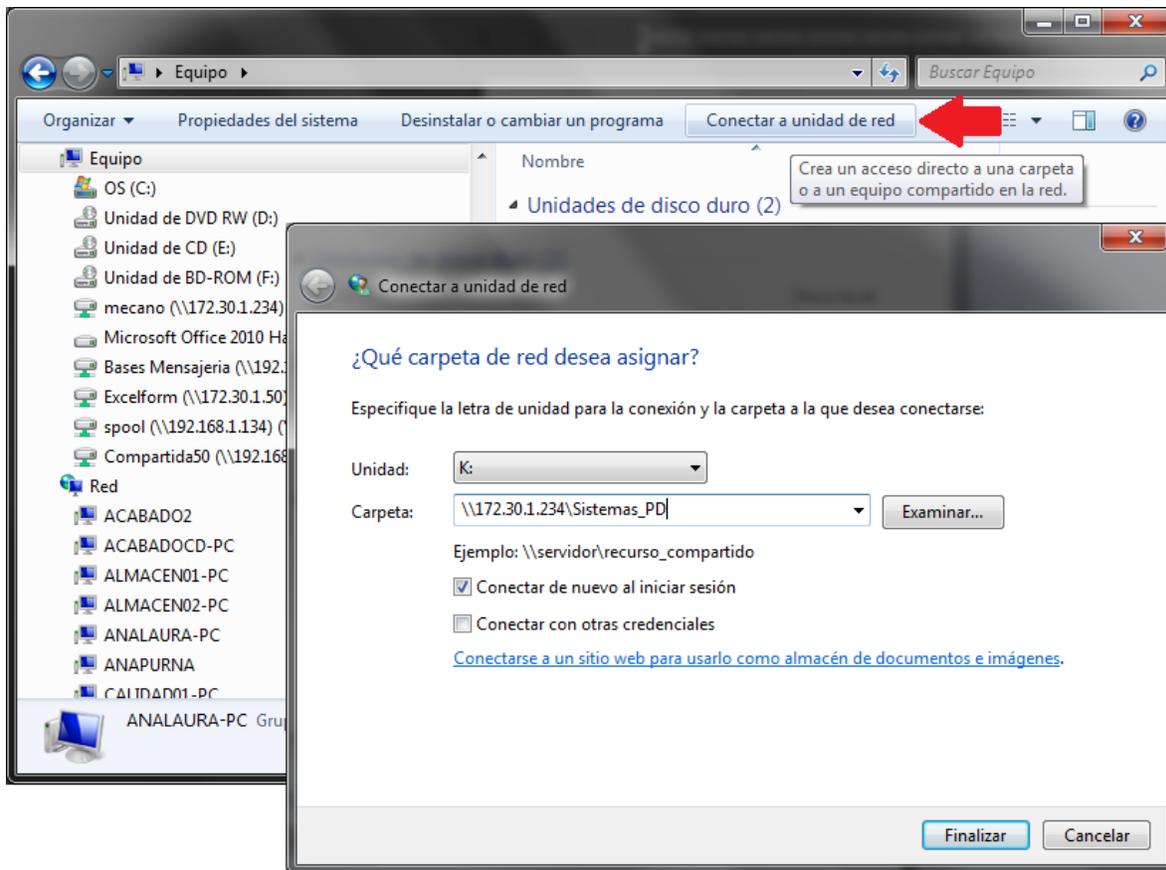


Figura 3.43 Configuración de la Unidad de Red.

- *Instalación de los DLL:* para su instalación dentro de C:\Windows\ en la carpeta System32 y Syswow64 (si es el caso) alojamos los archivos que se ven en la figura 3.44 y los instalamos utilizando el CMD escribiendo **regsvr32 "nombre del archivo".dll**

Nombre	Tamaño	Comprimido	Tipo	Modificado	CRC32
Folder					
..					
vfp9t.dll	3,878,912	1,817,114	Application Extension	13/12/2004 17:04 Hrs.	42A029D0
vfp9resn.dll	1,466,368	281,335	Application Extension	13/12/2004 17:10 Hrs.	EB6FF2F2
VFP9RENU.DLL	1,429,504	275,574	Application Extension	13/12/2004 15:36 Hrs.	8859A085
vfp9r.dll	4,710,400	2,268,291	Application Extension	13/12/2004 17:03 Hrs.	E315DDBF
msvcr71.dll	348,160	152,609	Application Extension	01/07/2013 11:39 Hrs.	35563170
MSCAL.OCX	111,472	50,848	ActiveX control	26/10/2008 15:58 Hrs.	0FA7BAC5
dllwin7.bat	239	161	Windows Batch File	30/07/2014 08:31 Hrs.	9D79A66B
dll.bat	312	146	Windows Batch File	30/07/2014 08:30 Hrs.	9033A456
CONFIG.FPW	14	14	Fichero FPW	11/02/2013 13:16 Hrs.	E1D40201
comcti32.ocx	609,824	200,686	ActiveX control	09/03/2004 16:45 Hrs.	3E45478A
Total 12,555,205 bytes en 10 ficheros					

Figura 3.44 DLL para instalación.

- *Instalación de Adobe PDF, Apache OpenOffice y establecer la comunicación del ODBC con la base de datos del nuevo sistema de cotizaciones. El nombre tiene que coincidir con el que se programa dentro del sistema, el código es el siguiente:*

```

cCadConnDSN = "dsn=MECA_SirPAG;uid=usuario;pwd=password"
nHanConn = Sqlstringconnect (m.cCadConnDSN)
IF nHanConn > 0
    SQLEXEC (nHanConn, 'SELECT * FROM dbo.ORDENMECANO WHERE
    N_orden='+PADL (lnOrden, 6, "000000"), 'TEMPORAL0')

```

Si la conexión fue exitosa podrá hacer la búsqueda de la O.T. de interés del usuario dentro de cursor y saber si existe o no y extraer la información que requiere la tabla OT para trabajar. En la figura 3.45 podemos ver el resultado de establecer la comunicación a través del DNS.

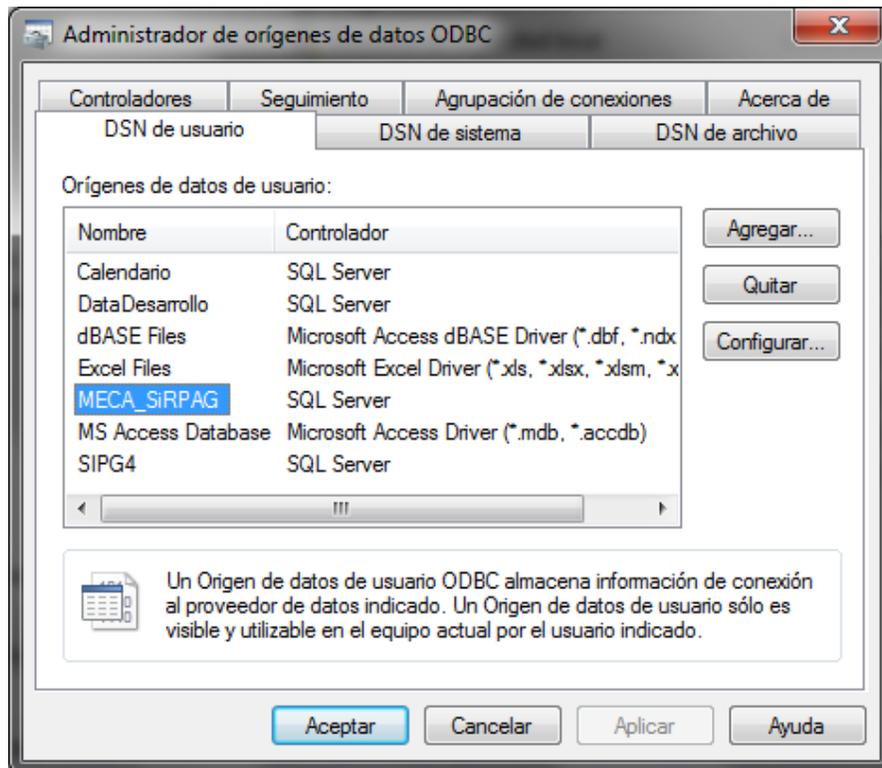


Figura 3.45 Nombre del ODBC.

- **Herramientas y entorno**

Además de utilizar Microsoft Visual FoxPro 9.0 con sus librerías predefinidas se utilizan dos elementos adicionales:

- Clase LMCAL.VCX

Como se puede apreciar en la figura 3.46 contiene un calendario que se muestra al hacer clic en el botón y tras seleccionar el día entrega ese valor al objeto que podrá asignarse a una variable.

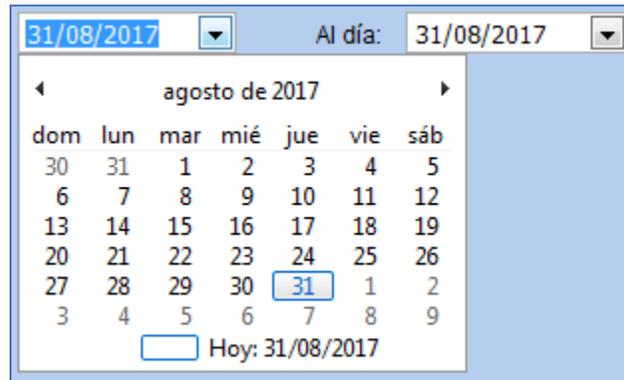


Figura 3.46 Interfaz del Objeto Calendario.

- Código fuente ThemedControls 3.5.8

Es una serie de Formas compuestas por clases que muestran una interfaz y componentes parecidos a los de Outlook 2003 proporcionadas por Emerson Reed en la página web <http://vfpx.codeplex.com> página aún vigente. Para dar una apariencia más amigable que podemos comprobar en la figura 3.47.

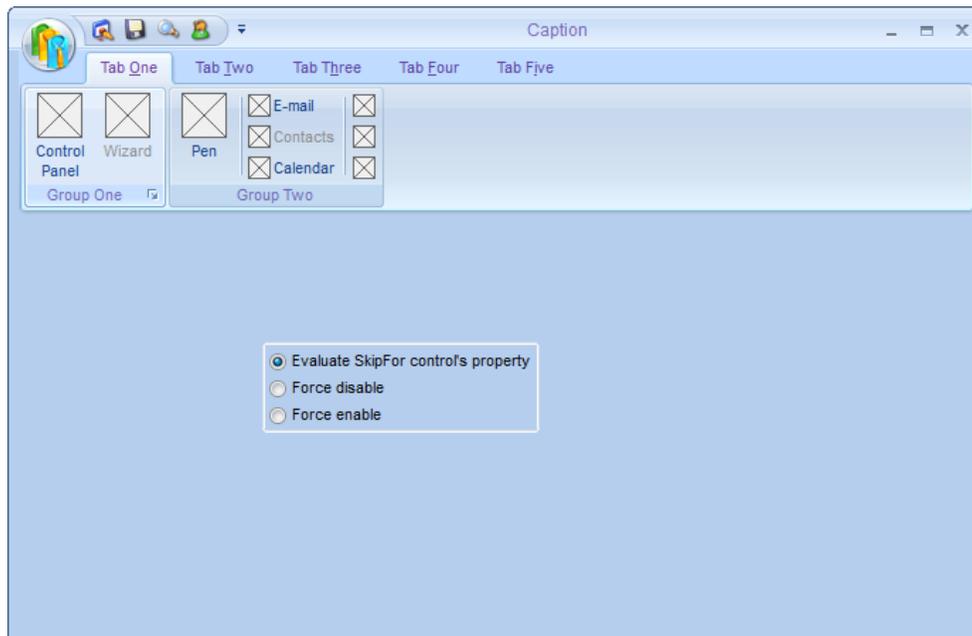


Figura 3.47 Interfaz del Código ThemedControls.

Al concluir la creación del sistema se construye un archivo tipo .exe (ejecutable) y no una app ya que el primero incluye el cargador de Visual FoxPro por lo que los usuarios no estarán obligados a tener instalado FoxPro en sus equipos de cómputo, sólo los .dll correspondientes que se mencionó en páginas atrás.

- **Capacitación del personal**

La implementación de estos dos métodos piloto y paralelo por parte del área de Correo Directo tendrá una duración de un mes para estudiar y apoyar en los por menores con la ayuda del departamento de Calidad y supervisores. Posterior se dará la capacitación del resto de los operadores de la planta introduciendo el sistema por el método paralelo por 3 meses para concluir con el uso de reportes escritos e introducir completamente el Sistema de Reportes.

La capacitación se realizó a grupos de 7 a 9 personas que pertenecieran a la misma área de producción para presentar los casos y ejemplos fáciles de comprender, realizando primero un ejemplo de reporte de producción por parte del exponente y otro ejemplo por parte de algún voluntario del grupo sin la intervención del instructor para que el resto del equipo le diera soporte, se compartían dudas y comentarios en un tiempo de 45 minutos el curso concluía, con dos cursos por día en una semana se concluyó la capacitación de todo el personal operativo y supervisores.

El Área de Calidad junto con el supervisor del área correspondiente revisaban los resultados del primer mes, tanto consultando los reportes electrónicos obtenidos por SiRPAG como los escritos a mano (método paralelo) orientando a los usuarios que requerían apoyo.

Antes de la liberación de SiRPAG para ser usado por el usuario final pasó por una serie de pruebas las cuales se describen a continuación.

- **Pruebas de unidad**

Las pruebas unitarias se realizaron en las acciones primarias de nuestro sistema según las figuras 3.11 a 3.16. Para que se cumplieran correctamente se restringió el ingreso de datos no válidos (como evitar caracteres alfabéticos en cuadros de texto que esperan un valor numérico etc.) pero este tipo de contramedidas se explicarán mejor en las Pruebas de Sistema.

- ***Abrir y cerrar reporte***

Posee una serie de condiciones para corroborar si cumple con `THISFORM.Newactivity()` y `THISFORM.Lastactivity()`. La variable `InBandera` determina si es un reporte nuevo o no, también posee condiciones IF para evitar errores y arrojar el mensaje correspondiente. La bitácora es primordial para conocer quién y cuándo se realizó una acción dentro del sistema.

- ***Agregar nueva O.T.***

Las variables dentro de la condición IF se les asignó un valor correspondiente al ingresado por parte del usuario dentro del formulario, aquí tenemos nuevamente el uso de IF para saber si podemos o no continuar, obligamos a que el usuario llene todos los campos que se requieren,

luego si el número de O.T. no se repite con algún otro existente permite continuar con el registro dentro de la tabla. Se manda a llamar a la bitácora y limpiamos las variables para evitar arrastrar algún dato que no necesitamos.

- ***Agregar máquina***

Con un botón se habilitan los campos para registrar una máquina y de igual forma guardarlo. Pero para poder habilitar los campos se necesita seleccionar el área a la que pertenecerá el nuevo equipo, el usuario cuando ingresa los datos no está consciente del valor asignado para cada tipo de unidad de medida, por ello en el listado ellos ven el concepto completo: extendido, revolución, etc. y el programa se encarga de asignar el valor que le corresponde. Después de guardar en la tabla actualiza el listado de máquinas que presenta el sistema al usuario.

- ***Consultar producción***

Las consultas se manejan por DO CASE. Dentro de la programación hay una bandera, inicialmente con el valor de 0, en cuanto elijan algún tipo de consulta se elige el tipo de query que se va a realizar a las tablas correspondientes.

- ***Guardar cambios en un reporte***

Cuando se busca un reporte de producción además de colocar el número de la O.T. se tienen que entregar los datos del usuario, máquina, fecha, actividad realizada y hora de registro. Cada que se selecciona una opción se va realizando un query cuyo resultado se vacía en la siguiente lista para que se busque correctamente el elemento. La variable IsFinal cargaría con la hora en que se registró el reporte. Si esta variable tiene un valor asignado muestra los campos que se pueden editar, así como permite guardar dichos cambios con la condición IF que lleva el código dentro del botón Guardar Cambios.

El método totaltime(tiempo inicial, tiempo final) calcula las horas y minutos que trabajó el operador de dicha actividad. Cuando se efectúan los cambios se guarda en bitácora y se limpia la pantalla, también limpiamos el contenido de la variable IsFinal para poder hacer cualquier otra modificación.

- ***Eliminar reporte***

La variable IsFinal carga con el valor de la hora de registro del reporte, con esto podemos constatar que se tiene un reporte seleccionado por parte del operador. Este cambio se guarda en la bitácora también.

- ***Agregar usuario***

La variable InPrioridad esta inicializada con el valor de 0, si el usuario registró a alguien del área de Impresión Variable o de Rígidos entonces tendrá una prioridad 1, le permitirá al nuevo usuario registrar más de un reporte a la vez en el sistema sin necesidad de cerrar una previa.

- **Eliminar usuario**

La variable IsName es el nombre del usuario del sistema seleccionado, cuando se elimine se guardan los cambios en la bitácora, en cada prueba unitaria se agregaron estas condiciones IF ELSE para evitar conflictos con los datos que espera el sistema procesar.

▪ **Pruebas de bases de datos**

La base de datos que contiene las tablas que utiliza SiRPAG tiene que ser abierta como tipo Compartida al igual que sus tablas, esto se hace porque son varios usuarios que están utilizando el sistema de manera simultánea, para guardar los cambios no pueden estar de manera privada, ya que sólo un usuario a la vez podría utilizar el sistema y otros tendrían que esperar su turno en toda la planta. Tras cargar todos los objetos que componen al formulario el siguiente proceso que se manda a llamar es el de Activar, allí es donde se abren las tablas:

```
IF !USED('SHINKA')
    OPEN DATABASE "K:\Sistema SiRPAG\Database\SHINKA.DBC" SHARED
ENDIF
IF !USED('USUARIO')
    USE USUARIO IN 0 ALIAS USUARIO SHARED
ENDIF
.
```

Sólo se realiza una vez esta acción, cuando se cierra la aplicación se cierra la base de datos. Dentro de la sección de base de datos en FoxPro se realiza la configuración que podemos ver en la figura 3.48 para la conexión con la base de datos de Mecano que comparte las O.T. desde la ODBC.

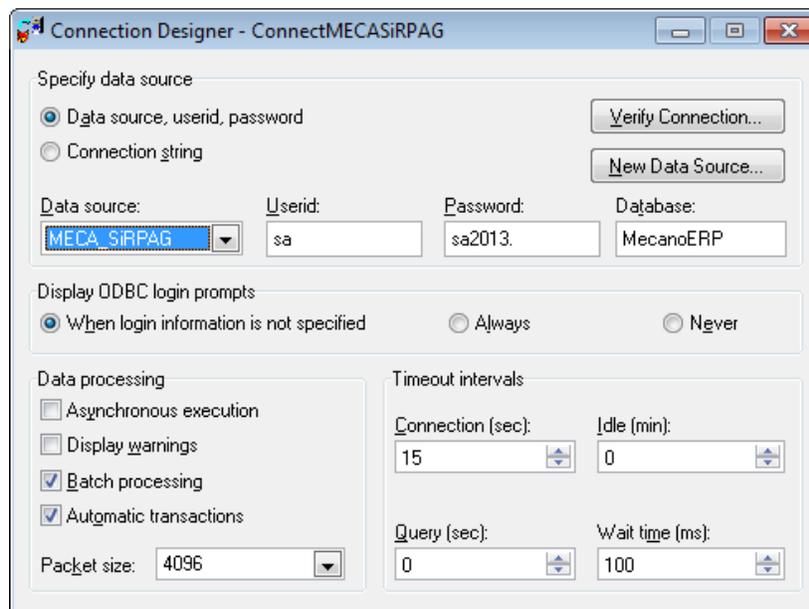


Figura 3.48 Configuración desde FoxPro para conectar con la ODBC.

- **Pruebas de seguridad y acceso**

```
intento = intento + 1
IF intento <= 5
    .
    .
    .
ELSE
    MESSAGEBOX("Sólo tiene 5 intentos, proceso cancelado.",16,"Login
    SiRPAG")
    THISFORM.Hide
    THISFORM.Release
    CLEAR EVENTS
    QUIT
ENDIF
```

Hay un contador que va guardando el número de intentos de acceso del usuario, si se equivoca más veces de lo autorizado cierra la aplicación. Para acceder se requiere de un usuario y una contraseña, la contraseña está limitada a sólo números y letras del alfabeto, esta restricción es una de las tantas que se agregó dentro de los componentes del formulario descritos dentro de las pruebas del sistema, se añadieron desde la construcción del objeto para evitar escribir código adicional. Como se puede ver, se almacena la hora de acceso también dentro de la bitácora, otro componente importante que se guarda en la bitácora es el nombre del equipo desde donde se trabajó, todas estas medidas se usan para cerciorarnos dónde se registró alguna actividad, si ocurrieron anomalías o errores la misma bitácora lo indicará.

### 3.4 Integración

- **Pruebas de sistema**

Para restringir lo que ingresan los usuarios dentro de SiRPAG se realizaron algunas acciones, desde el inicio de sesión para evitar que ingresen mal su nombre de usuario lo podrán buscar desde un listado que ya se alimenta de manera directa de la tabla de USUARIO, para que sólo escriban su contraseña.

En las cajas de texto dentro de la aplicación sólo reciben el tipo deseado, por ejemplo, para ingresar el número de O.T. al que va a pertenecer el reporte la caja de texto no permite recibir nada más que números, sin espacio u otros caracteres, la configuración se hace desde las propiedades del objeto como lo muestra la figura 3.49.

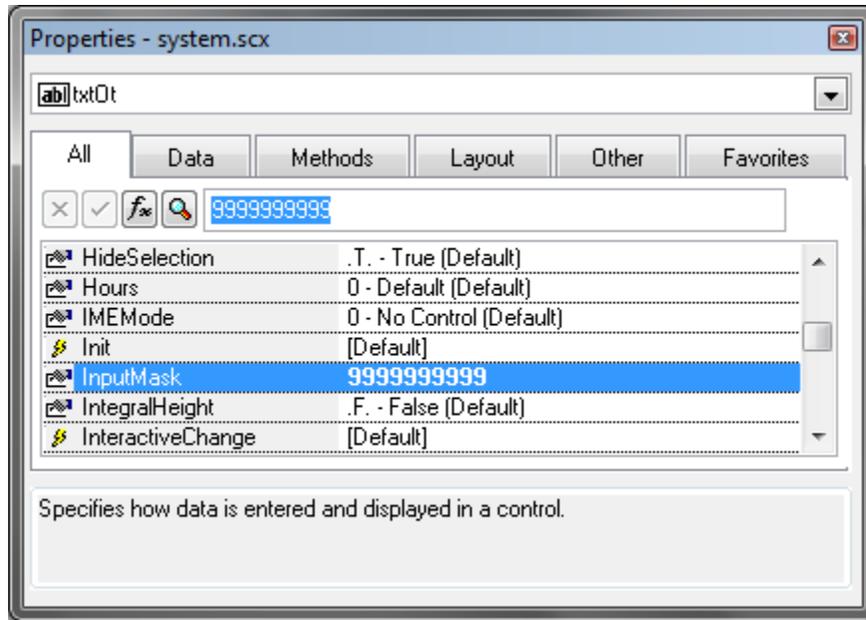


Figura 3.49 Propiedades de un componente del formulario.

Otra manera de configurar la forma en que recibe la información el sistema es inicializando la caja de texto desde que se muestra al usuario, en el caso de los horarios en la Edición de Reportes se ve de la siguiente forma: 20/04/2017 17:55:37 Hrs. De esta manera, cuando el usuario edita la información, si no cuenta con la estructura que el objeto ya le está mostrando no le permite avanzar, cosa que podemos ver también con la selección del calendario en la Consulta de Producción.

Si observamos nuevamente la figura 3.31 vemos que hay demasiadas ventanas de alerta e información, cuyo objetivo es orientar al usuario si está cometiendo algún error u omisión de la información evitando que proceda a almacenar o modificar de manera errónea.

- **Verificación y validación**

Tras realizar una revisión exhausta de los posibles contratiempos que ya se mitigaron en las pruebas realizadas en los temas anteriores, la *verificación* como ya definimos es saber si las partes de la programación que procesan los datos de entrada son los correctos, se evaluaron las posibles entradas que daría un usuario, los caminos y decisiones y además de las propiedades definidas en los objetos del formulario se añadieron códigos de programación para cualquier imprevisto.

Por ejemplo, hay columnas dentro de la tabla REPORTE que esperan un valor numérico, si no recibe alguno entonces se almacena como 0, por lo que si el usuario no añadió ningún valor esperado el sistema pone por default el 0, otro ejemplo es que no reconozca el tipo de valor que esté ingresando en la caja de texto, con el comando VARTYPE se puede saber si el contenido de la caja es numérico o decimal o cadena de caracteres.

```

IF !EMPTY(...txtTotal.Value)    &&Cantidad
    lsCheck = VARTYPE(...txtTotal.Value)
    IF lsCheck == "N"
        lnCantidad = ...txtTotal.Value
    ELSE
        lnCantidad = INT(VAL(...txtTotal.Value))
    ENDIF
ELSE
    lnCantidad = 0
ENDIF

```

La verificación pasa desde elementos de entrada ingresados en el sistema a los resultados obtenidos, así implementando cada tipo de prueba descrita y su correspondiente solución (explicada en los puntos anteriores) para realizar una nueva prueba con datos de entrada semejantes para ver que el resultado sea el correcto.

En la validación se comprueban que los requerimientos estén presentes sin falta en el diseño y desarrollo, comprobables para el usuario final mostrando que se construyó de manera correcta el sistema. Para evitar duplicidad en la tabla OT que podría causar una serie de problemas a la hora de ejecutar las consultas de producción fue que al intentar registrar una O.T. de manera manual en el apartado de Nueva O.T. primero el sistema se cerciora que no exista el número dentro de la tabla OT.

### 3.5 Funcionamiento

- **Pruebas de entorno**

Las pruebas de entorno nos permiten verificar que el sistema fluye de manera correcta en los equipos donde se instaló. Para que el sistema corriera de manera simultánea en todos los equipos se cumplió con una condición explicada en las pruebas en las bases de datos. En la figura 3.50 vemos arriba el sistema corriendo en una PC de 2 GB de RAM y un procesador Intel Pentium 4 CPU 2.80GHz y en la segunda la aplicación corriendo en una PC de 1 GB de RAM y Procesador Intel Pentium 4.

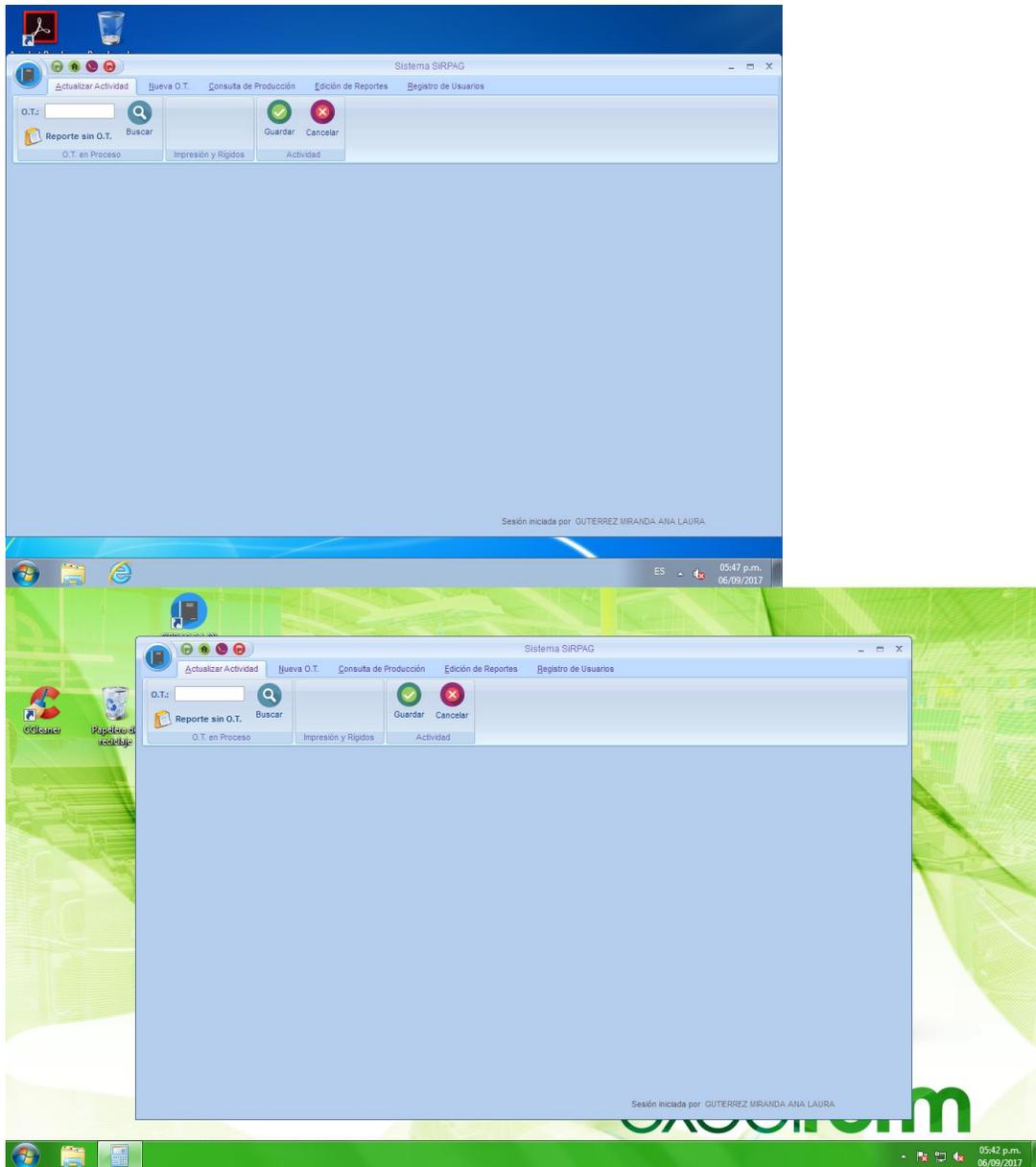


Figura 3.50 SiRPAG en equipos distintos.

- **Mantenimiento**

El mantenimiento preventivo consta en hacer pruebas de caja negra ingresando datos ficticios desde una cuenta de usuario que no pertenezca a la de producción para no influir en los registros de los operadores, realizamos las consultas de lo que se almacenó y comparamos lo que se ingresó contra lo que se obtuvo. De la misma manera con la edición y eliminación de reportes (de usuarios ficticios).

El mantenimiento perfectivo se realizará en cuanto se libere por completo el nuevo sistema de cotizaciones, ya que la tabla OrdenMecano alimentará la tabla OT. Así quedando esta pestaña sólo para agregar y eliminar registros de máquinas de la empresa.

El mantenimiento adaptativo se presentaría en el caso de que se abriera un área nueva en la planta tendría que hacerse ajustes a la tabla que alimenta las áreas de producción, así como los niveles de privilegios llegaran a cambiar, pero estos ajustes serían dentro de la base de datos.

El mantenimiento correctivo se realizó y supervisó a partir de las pruebas y puntos mencionados en las páginas anteriores.

Para la documentación del mantenimiento se cumplen con los puntos solicitados en los requerimientos del desarrollador para Verificar que todo se encuentre estable:

- Ya sea mantenimiento preventivo, perfectivo, adaptativo y correctivo no afectará al resto de las operaciones que podrán seguir en funcionamiento dentro del horario acordado.
- La documentación describirá los siguientes puntos:
  - o Motivo del mantenimiento
  - o Firma de autorización
  - o Descripción del mantenimiento
  - o Descripción técnica: operaciones, movimientos y líneas de código
  - o Conclusiones y consideraciones especiales
- Monitoreo

Las actividades fundamentales de la metodología en cascada modificada descritas en el capítulo 2 se pueden ver ampliamente en este capítulo.

La toma de los requerimientos es junto con el diseño de los procesos más largos y metódicos como podemos ver, ya que la implementación, integración y funcionamiento si se tiene los recursos humanos y tecnológicos al alcance, fluye de manera progresiva.

En la vida laboral no siempre se va a contar con la tecnología de nuestra preferencia para desarrollar, nos tenemos que ajustar a un panorama diferente, pero lo que siempre se va a mantener igual es la metodología, sea cual sea en el lenguaje de programación o entorno de desarrollo la metodología dicta la forma en que se trabajará.

El mantenimiento a pesar de ser la última de las actividades de la metodología, no deja de ser menos importante, si no se toma en cuenta los elementos que conforman el diseño y las precauciones según sea el tipo de mantenimiento el sistema podría perder su funcionalidad.

# **Capítulo 4**

## **Resultados, impacto y conclusiones**

## 4.1 Resultados

### Cumplimiento de los requerimientos

Con el producto terminado se demostró a la compañía la magnífica herramienta que puede llegar a ser el desarrollo de aplicaciones para facilitar trabajos en la empresa. El llenado de reportes de producción que se volvió una tarea tardía por parte del operador (y sin ningún argumento fiable de si es correcta o no la información) y siendo recopilada por contadas personas a lo largo del día que también podrían cometer equivocaciones a la hora de calcular el tiempo total de trabajo del operador, SiRPAG llenó las expectativas de una compañía que veía hasta hace poco un fragmento de todo lo que ocurre en la planta productiva.

#### 1. Comunicación entre los principales sistemas de información

Ya que el anterior sistema de generación de pedidos es un software comercial no era posible modificar su código para extraer de su base de datos la información por lo que se fabricó un apartado en SiRPAG para dar de alta una nueva O.T. y poder hacer sus reportes correspondientes:

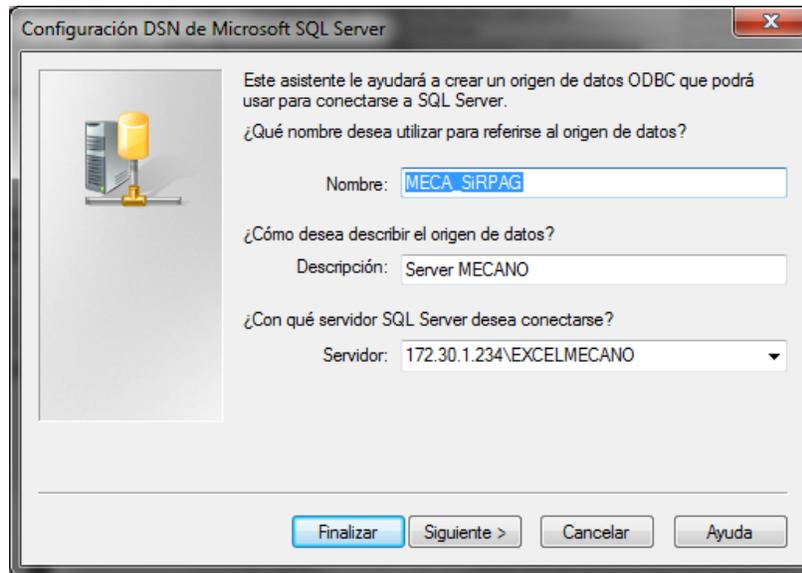


The screenshot displays the SiRPAG web application interface. At the top right, it is labeled 'Sistema SiRPAG'. A horizontal navigation bar contains several menu items: 'Actualizar Actividad', 'Nueva O.T.', 'Consulta de Producción', 'Edición de Reportes', and 'Registro de Usuarios'. Below this bar, a floating toolbar is visible with three buttons: 'Nueva O.T.' (with a plus icon), 'Guardar' (with a checkmark icon), and 'Cancelar' (with an X icon). The main content area features a form with the following fields: 'O.T.' with the value '2007', 'Pedido' with '1200', 'Partida' with '2', 'Cliente' with a dropdown menu showing 'APOYO TELEFONICO CABLEMAS, S.A. DE', and 'Aplicación' with 'CARTA 2DO TRIMESTRE'.

La información ingresada por el usuario en esta pestaña se almacena en la tabla OT para que pueda ser referenciada en los reportes operativos.

En el caso de las O.T. generadas en el nuevo sistema de cotizaciones, SiRPAG se conecta a una tabla de Mecano para obtener la información de la O.T. y almacenarla de manera automática cuando un operador registra por primera vez un reporte con ese número de O.T. dentro del SiRPAG.

En FoxPro nos podemos conectar a una base externa a través de una DSN ya configurada con los datos de tipo de controlador, servidor, instancia, base de datos, nombre y contraseña de acceso:



Vemos en la primera línea de código que primero se escribe el nombre de la DSN creada y el nombre de administrador y contraseña para acceder a la instancia de Mecano. En la segunda línea establece la conexión de tipo SQL, cuando la comunicación se realiza de manera exitosa (cumple la condición IF), hace la consulta de la información de la O.T. ingresada por el operador y la almacena de un cursor.

La información de la O.T. registrada en Mecano se almacena en la tabla OT:

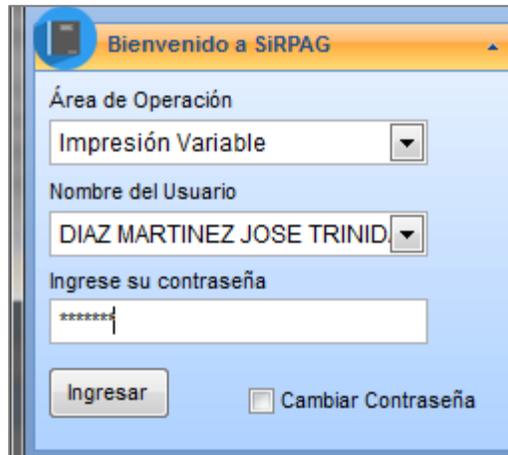
Orden	Pedido	Partida	Ciente	Aplicacion
95	8042	1	INDUSTRIAS AUDIOVISUALES MEXICANAS,	BROCHURE MOANA
96	8039	1	BANCO INMOBILIARIO MEXICANO	ESTADOS DE CUENTA
99	8040	1	PASTELERIA Y CONFITERIA	IMPRESIONES A COLOR
100	8043	1	INTERCAM	PATRIMONIAL BANCO
103	8043	4	INTERCAM BANCO S.A. INSTITUCION DE	D MENSAJERIA ESPECIALIZADA ORDINARIA
104	8049	3	BANCO DEL BAJIO, S.A.	CABALLETE BILLETERA MOVIL
105	9157	1	EXCEL FORM	TARJETA DE PRESENTACION
108	8047	3	BANCO DEL BAJIO, S.A.	CARTA BIENVENIDO " ESTIMADO CLIENTE "
109	8050	1	SANBORN HERMANOS, S.A.	MANTELETA
110	8046	1	SHEAFFER DE MEXICO, S DE RL DE CV	GARANTIA 1 AÑO
111	8054	1	PROVIDENT	CONTRATOS PF-MK-NA-02-01
112	8052	1	ELECTRONIC BILLS & SERVICES, S.A. D	DIPTICO FONDO INTERS1 ENERO 2017
113	8057	1	MAREL DE MEXICO, S.A. DE C.V.	NUEVA CAJA PUSH-UP
114	8058	1	GANADEROS PRODUCTORES DE LECHE PURA	PAPEL STOCK MEDIA CARTA 9 1/2 X 5 1/2 -I TANTO CO

La próxima ocasión que se realice un reporte de una O.T. que ya existe en la tabla OT no será necesario entablar la comunicación con Mecano ni dar de alta en la pestaña Nueva O.T.

## 2. Ventana inicial que solicite al usuario su nombre y contraseña para autenticar que sea un empleado de la empresa e inicie sesión

Para evitar estar escribiendo el nombre completo cada vez que se inicie sesión se creó un listado donde podrán seleccionar su nombre y un cuadro de texto para escribir su contraseña. Como son

varios los empleados registrados en el Sistema hay un listado previo de las áreas de trabajo que servirá de filtro para poder encontrar su nombre:

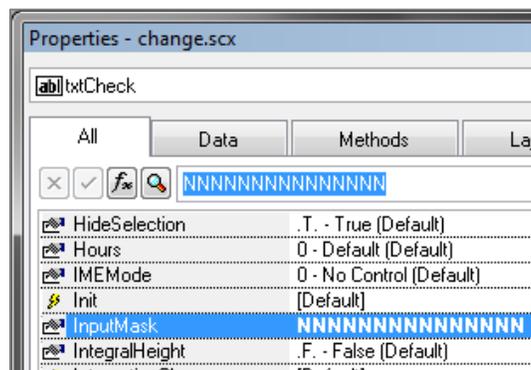


Para que nadie más vea el ingreso de la contraseña se configuró en la máscara del textbox sólo mostrar asteriscos. Si el usuario no se localiza en los listados significa que no ha sido registrado aún. El primer listado se alimenta de la tabla USUARIO de la columna Labor.

Guarda en una variable el área de trabajo seleccionada, con ese dato se realiza la búsqueda de todos los trabajadores que pertenecen al área dentro de la tabla USUARIO, lo almacena en un cursor que posteriormente se vaciará en el siguiente combobox. Cuando todos los datos son llenados de manera correcta al hacer clic al botón de Ingresar activa el método click() que valida la contraseña e inicia la sesión de SiRPAG.

### 3. Una opción para poder hacer cambio de su contraseña y consultar su historial

Si el usuario desea cambiar su contraseña predeterminada (número de empleado) se encuentra la opción de cambiar la contraseña, la máscara del textbox fue configurada para sólo recibir un máximo de 15 números y/o letras.



Para la consulta del historial personal sólo el usuario podrá ver su trabajo según lo que elija en el

calendario, no hay manera de ver el de alguien más ya que la consulta de la tabla es entre la fecha inicial y final y el nombre del que inicie la sesión.

O.T.	Producto	Fecha	Máquina	Clave	Hora Inicial
1790	CONTRATOS PF-MK-NA-02-01	06/07/2017	Printronix 7215	PRODUCCIÓN	06/07/2017 19:26:11 Hrs.
1790	CONTRATOS PF-MK-NA-02-01	06/07/2017	Printronix 8220	PRODUCCIÓN	06/07/2017 21:21:06 Hrs.
1790	CONTRATOS PF-MK-NA-02-01	07/07/2017	Printronix 8220	PRODUCCIÓN	07/07/2017 00:01:41 Hrs.
1790	CONTRATOS PF-MK-NA-02-01	07/07/2017	Printronix 7215	PRODUCCIÓN	07/07/2017 00:21:44 Hrs.
1790	CONTRATOS PF-MK-NA-02-01	07/07/2017	Printronix 8000	PRODUCCIÓN	07/07/2017 00:43:09 Hrs.
1911	ELITE	07/07/2017	Ricoh Pro C751	ARREGLO DE MAQUINA	07/07/2017 19:06:47 Hrs.
1911	ELITE	07/07/2017	Ricoh Pro C751	PRODUCCIÓN	07/07/2017 19:19:34 Hrs.
1915	CARTA ART 61	07/07/2017	Ricoh Pro 8120	ARREGLO DE MAQUINA	07/07/2017 21:53:42 Hrs.
1915	CARTA ART 61	07/07/2017	Ricoh Pro 8120	PRODUCCIÓN	07/07/2017 21:58:48 Hrs.
1790	CONTRATOS PF-MK-NA-02-01	08/07/2017	Printronix 8000	PRODUCCIÓN	08/07/2017 00:03:56 Hrs.
1943	CASA DE BOLSA	10/07/2017	Ricoh Pro 8120	VOBO	10/07/2017 15:52:43 Hrs.
1943	CASA DE BOLSA	10/07/2017	Oce Vario Print 120	VOBO	10/07/2017 15:53:02 Hrs.
1943	CASA DE BOLSA	10/07/2017	Oce Vario Print 120	PRODUCCIÓN	10/07/2017 16:11:39 Hrs.
1943	CASA DE BOLSA	11/07/2017	Oce Vario Print 120	PRODUCCIÓN	11/07/2017 08:23:56 Hrs.
1952	CARTA COBRANZA	11/07/2017	Oce Vario Print 120	VOBO	11/07/2017 12:44:29 Hrs.
1952	CARTA COBRANZA	11/07/2017	Oce Vario Print 120	RIPEO DE ARCHIVO	11/07/2017 12:50:08 Hrs.
1952	CARTA COBRANZA	11/07/2017	Oce Vario Print 120	PRODUCCIÓN	11/07/2017 12:56:04 Hrs.
1955	CARTA COBRANZA	11/07/2017	Oce Vario Print 120	VOBO	11/07/2017 13:39:00 Hrs.
1958	FONDOS	11/07/2017	Oce Vario Print 120	VOBO	11/07/2017 16:35:20 Hrs.
1958	FONDOS	11/07/2017	Oce Vario Print 120	RIPEO DE ARCHIVO	11/07/2017 16:46:06 Hrs.

Sesión iniciada por HERNANDEZ MORENO SUSANA ITZEL

4. Contener una pestaña con la opción de **Actualizar Actividades** que estén relacionadas a una O.T. o adicionales, cada que genere una nueva actividad tendrá que cerrar la anterior excepto para Impresión Variable y Rígidos quienes usan más de un equipo a la vez. Habrá campos que se llenarán de forma automática y si el usuario no cierra su reporte no será considerado su tiempo de producción y tendrá que solicitar el cambio a Calidad

Cuando el operador realiza sus actividades al reporte se le asigna un número de O.T. o el 999999 para la categoría de procesos adicionales fuera de una O.T.:



Si es una O.T. se ingresa el número en el textbox y da clic en el botón Buscar, para tareas adicionales es directamente con el botón inferior. Los puntos obligatorios son la máquina, y actividad que realizará:

Nueva Actividad:

Área:  Máquina:

Comentarios:

Máquina:

- Pagestream Océ 372-1
- Pagestream Océ 372-2
- Printronix 5215
- Printronix 7215
- Printronix 8000
- Printronix 8220
- Ricoh Pro 8120

En el caso de ser un operador del área de Impresión Variable o de Rígidos podrá agregar una nueva actividad de la misma O.T. o cerrar la anterior si lo requiere:



Esto se determina a partir de la tabla USUARIO si en la columna Prioridad poseen el valor 1.

Si eligen cerrar la actividad anterior tendrá que acceder como la primera vez y guardar los cambios si aplica, y si desea agregar una nueva actividad a la misma O.T. podrá llenar su nuevo reporte en la parte inferior:

Máquina Utilizada:

Actividad Anterior:  Cantidad terminada:  Página(s) "PDF":

Comentarios:

Nueva Actividad:  Acabado en:

Área:  Máquina:

Comentarios:

Acabado en:

- ENGOMADO
- CLASIFICACIÓN
- EMPAQUE
- ENGRAPADO
- RECIBIR EN MAQUINA
- SELLADO MANUAL
- EMBOLSADO MANUAL

Si la actividad ACABADO Y/O APOYO OTRAS ACTIVIDADES es seleccionada se habilitará un listado de los diferentes tipos de acabados para dar un informe más detallado. La fecha, hora, nombre (variable global compartida desde el inicio de sesión) del operador y tiempo total de la actividad los ingresa SiRPAG en automático con las instrucciones: `DATE ()` , `DATETIME ()` , `THISFORM.totaltime ()` .

Para Rígidos se habilitan nuevos campos al llenar sus reportes que son el Largo, Ancho y Tipo de Material y al cerrar el número de pasadas. Y si la actividad que concluirán es la Preparación de tinta se muestran los tipos de tintas:

Máquina Utilizada: Anapurna M2500

Actividad Anterior: PREPARACIÓN DE TINTA

Comentarios:

Nueva Actividad: ARREGLO DE MAQUINA

Área: Rígidos Máquina: Plotter HP Latex 570

Comentarios:

Largo: 6 Ancho: 7 Tipo de Material: Coroplast

# de Pasadas: 5 Cyan: 0.1 Magenta: 0.34 Amarillo: 3.1

Negro: 0.1 Light Magenta: 0.0 Light Cyan: 0.023

Para calcular el tiempo que tomó realizar una actividad se usó el método `totaltime()`. Si algún operador no cierre su reporte al día siguiente cuando se realicen las consultas de producción se cerrarán todas las actividades incompletas de manera automática y sin ninguna contabilización de tiempo productivo con el método `THISFORM.Closingprocess()`.

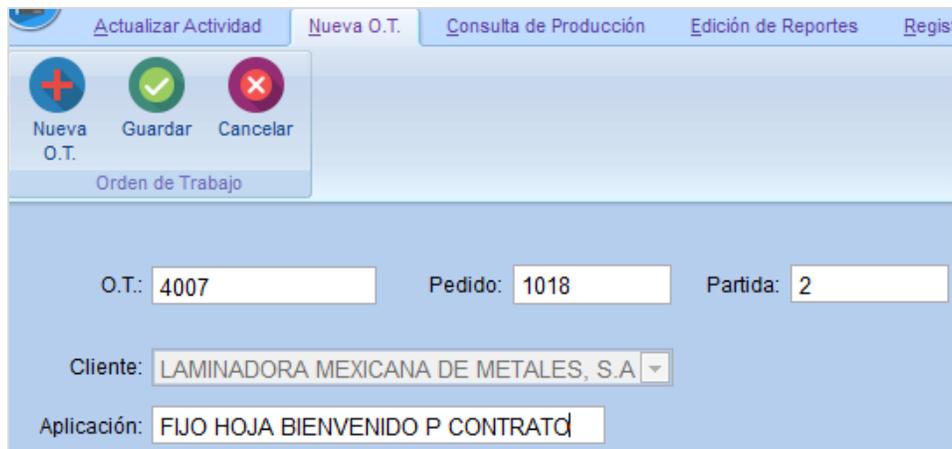
Para saber cuáles serán los reportes que se cerrarán el método obtiene la fecha actual del sistema y se eligen los reportes escritos de 30 días atrás hasta los de un día antes y que no tengan nada registrado en la hora final.

Cuando un usuario olvidó registrar la conclusión de su actividad al día siguiente aparecerá como actividad incompleta, no se podrá registrar cantidad de producción y tendrá guardado en la sección de Coment el mensaje que indica que olvidó cerrar su actividad.

Horainicio	Horafinal	Cantidad	Coment	Tiempo
28/09/2017 12:00	28/09/2017 13:01	322	SEPARAR POR DIA 10 PAQS. DE LUNES 7 PAQS. DE MAR	1:1
28/09/2017 12:03	28/09/2017 14:04	24	ARMAR PAQ. DE 20 X DIA 17 PAQ. EMPLAYAR 7 PAQ. Y E	2:1
28/09/2017 12:09	28/09/2017 12:09		Tiempo no contabilizado, el Usuario olvidó cerrar su actividad	0:00
28/09/2017 12:12	28/09/2017 15:30	900	LAMINADO DE FTRENTE Y VUELTA	3:18
28/09/2017 12:29	28/09/2017 12:29		Tiempo no contabilizado, el Usuario olvidó cerrar su actividad	0:00
28/09/2017 12:29	28/09/2017 12:29		Tiempo no contabilizado, el Usuario olvidó cerrar su actividad	0:00
28/09/2017 12:31	28/09/2017 12:31		Tiempo no contabilizado, el Usuario olvidó cerrar su actividad	0:00

**5. Contener una pestaña con la opción de Nueva Orden de Trabajo que permita dar de alta las O.T. de manera manual y dar de alta y baja las máquinas de la Empresa**

En esta pestaña se puede agregar una nueva O.T. al dar clic en Nueva O.T. y llenando los datos de O.T., pedido, partida, cliente y aplicación. Se puede seleccionar el cliente y la aplicación de los combobox que se alimentan de la tabla OT o agregar desde el teclado el nuevo elemento en los textbox que aparecen al dar clic a los botones amarillos (después de dar clic desaparecerán) que se encuentran a lado de los combobox:



Para agregar los datos de una nueva máquina se requiere dar clic igualmente en Nueva O.T. y seleccionar el área donde se localiza el equipo, después dar clic en el botón amarillo localizado a la derecha del combobox de Máquina y llenar los datos que se piden:



Para eliminar de la tabla MAQUINA algún elemento, se selecciona el área operativa y la máquina de los combobox y se da clic en el botón de una ( X ) a la derecha del combobox de Máquina.

**6. Contener una pestaña con la opción de Consulta de Producción que permita mostrar y descargar los tiempos de producción por O.T., producto, máquina y operador**

Para poder habilitar la opción de consulta se selecciona alguna de las cuatro opciones disponibles, cada una posee un número del 1 al 4. Al ser seleccionados el tipo de búsqueda se mostrará la información que tendrá que llenar.

Muestra los label y combobox en la pantalla con el método `Page03groupa()`, mientras los label tendrán como título lo que `lstTitulo01` y `lstTitulo02` contengan y el primer combobox será llenado de lo que el cursor Temporal posea.

Área: Impresion digital    Del día: 01/06/2017    Al día: 02/10/2017  
 Máquina: Oce Vario Print 120

O.T.	Producto	Operador	Fecha	Máquina	Clave	Hora Inicial
1538	CASA DE BOLSA	HERNANDEZ MORENO SUSANA ITZEL	09/06/2017	Oce Vario Print 120	ARREGLO DE MAQUIN	09/06/2017 09:...
1538	CASA DE BOLSA	HERNANDEZ MORENO SUSANA ITZEL	09/06/2017	Oce Vario Print 120	RIPEO DE ARCHIVO	09/06/2017 09:...
1538	CASA DE BOLSA	HERNANDEZ MORENO SUSANA ITZEL	09/06/2017	Oce Vario Print 120	PRODUCCIÓN	09/06/2017 09:...
1547	SIF	HERNANDEZ MORENO SUSANA ITZEL	10/06/2017	Oce Vario Print 120	PRODUCCIÓN	10/06/2017 07:...

Al seleccionar el botón Generar se muestra el resultado de la consulta en la ventana y se podrá descargar al dar clic en el botón Descargar y seleccionar el destino del archivo.

orden	aplicacion	operador	fecha	maquina	clave	horainicio	horafinal	cantidad	coment	tiempo
1462	INTERACCIO	JUAN ROJAS	02/06/2017	Oce Vario Pri	PRODUCCIÓN	02/06/2017 20:43	02/06/2017 22:56	5090		2:13
1463	INTERACCIO	JUAN ROJAS	02/06/2017	Oce Vario Pri	PRODUCCIÓN	02/06/2017 22:57	02/06/2017 22:57	0	Tiempo no cc	0:00
1463	INTERACCIO	JUAN ROJAS	03/06/2017	Oce Vario Pri	PRODUCCIÓN	03/06/2017 00:09	03/06/2017 02:54	21448		2:45
1465	BANCO-CHE	JUAN ROJAS	03/06/2017	Oce Vario Pri	PRODUCCIÓN	03/06/2017 02:55	03/06/2017 02:55	0	Tiempo no cc	0:00

### 7. Contener una pestaña con la opción de Edición de Reportes que permita corregir los reportes creados dentro del sistema o eliminarlos con sólo dar los datos del número de O.T., usuario, máquina, fecha, actividad y hora inicial

Se busca que existan primeramente reportes con la O.T. ingresada, al seleccionar el botón de Buscar a la derecha del textbox habilitará el resto de los campos a llenar si existe algún reporte.

O.T.: 41007   
 Usuario: MIRANDA HERNANDEZ    Máquina: Bostitch    Fecha: 24/10/2016    Actividad: PREPARAR    Hora Inicial:

Cada vez que se elija algún elemento en los combobox irán actualizando el siguiente combobox con las instrucciones dentro del método `click()`

En el caso del combobox de Fecha su contenido será todas las fechas de todos los reportes que posean el número de O.T. que se haya ingresado en el textbox que estén a nombre del elemento seleccionado en el combobox del Usuario y máquina seleccionada en el combobox Máquina. Después de llenar todos los datos se vuelve a seleccionar el botón de Buscar y habilitará todos los campos editables del reporte.

O.T.: 41007   
 Usuario: MIRANDA HERNANDEZ    Máquina: Bostitch    Fecha: 24/10/2016    Actividad: PREPARAR    Hora Inicial: 24/10/2016  
 Área: Acabado Datos Variables  
 Máquina: Embolsadora SPEED BAG  
 Hora Inicial: 24/10/2016 13:22:18 Hrs    Hora Final: 24/10/2016 15:22:18 Hrs  
 Cantidad: 2100

Para guardar la información se da clic en el botón Guardar y se llama al método `totaltime()` y se calcula nuevamente las horas y minutos del nuevo reporte. O si se requiere eliminar el reporte por completo se selecciona el botón Eliminar.

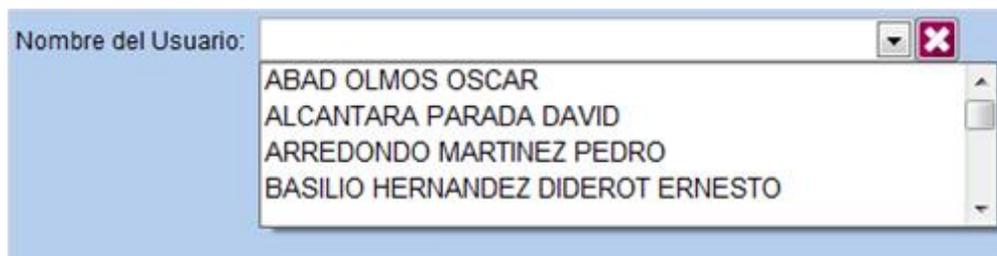
**8. Contener una pestaña con la opción de Registro de Nuevos Usuarios que permitirán dar de alta y de baja los usuarios de SiRPAG**

Con el botón Nuevo Usuario habilita los campos requeridos para dar de alta al nuevo usuario, como se describe a la derecha de los botones superiores de la pestaña, cada letra que contiene el combobox de Nivel representa el nivel de privilegios que poseerá el usuario dentro de SiRPAG.



Para el llenado de los combos se usa cursores que almacenan consultas sobre las tablas. Para hacer el llenado se aclara que el combobox se llenará a partir de un alias y posteriormente el nombre de ese alias.

En el caso de eliminar a algún usuario, de la misma manera el combobox se llena a partir de cursor compuesto por la consulta de todos los usuarios de la tabla USUARIO.



Después de seleccionar al usuario se selecciona el botón a lado derecho del listado.

**9. Se tiene que ocupar el software y hardware que ya tiene la empresa para el desarrollo e implementación de SiRPAG. Usando Visual FoxPro 9.0 en el software y diversos PC que hay en la empresa**

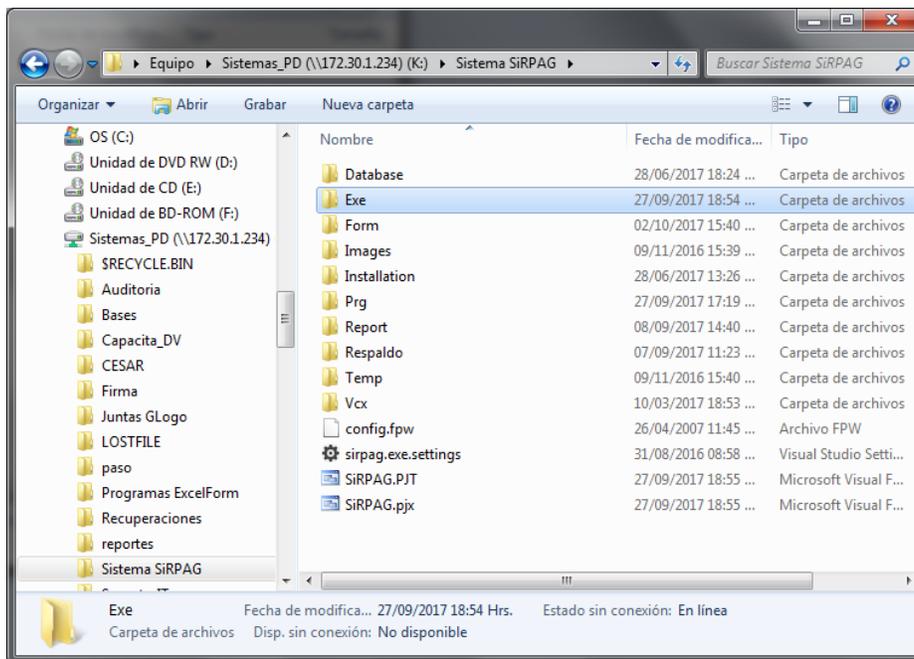
El único software de desarrollo de aplicaciones que posee licencia la empresa es el de Visual FoxPro 9.0 cuya vista predeterminada de la interfaz en los formularios es la siguiente:



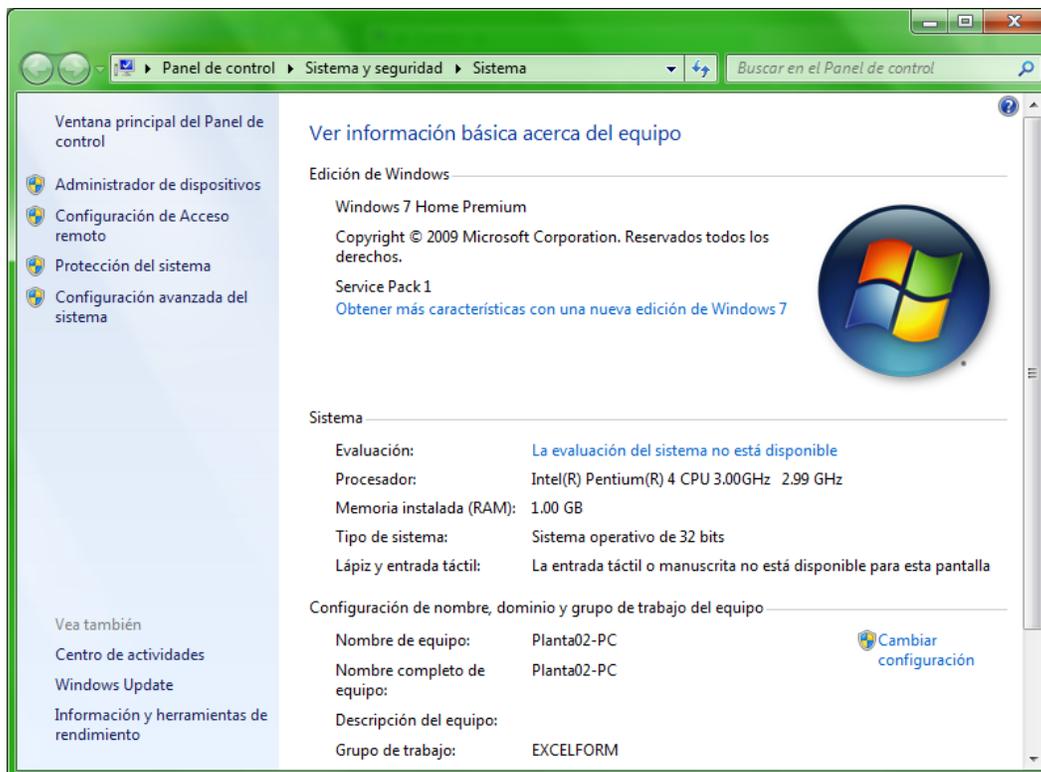
El cliente solicitó una interfaz más presentable por lo que se trabajó con código compartido (sin fines lucrativos) a través de Internet para mostrar una nueva interfaz además de diseños y plantillas (descarga gratuita) para los íconos y portada:



Alojamiento del sistema SiRPAG en el servidor Microsoft Windows Server 2012 y colocar sólo accesos directos en los equipos donde se trabajen, dejar la unidad de red donde se localizan las aplicaciones de la empresa con la Unidad K:



Las aplicaciones en FoxPro son compatibles con sistemas de Microsoft Windows de 32 bits. La gran parte de las PC donde se colocó el acceso al sistema poseen las siguientes características que las hacen compatibles con SiRPAG:



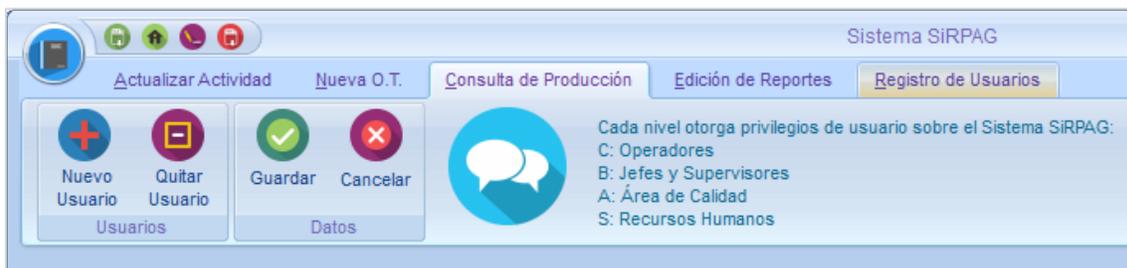
**10. Según sea la pestaña de SiRPAG sólo personal autorizado tendrá acceso a ella a través de una clasificación de letras: C, B, A y S**

Para validar la restricción de pestañas en SiRPAG desde que inicie sesión el usuario se agregó el método activate().

La instrucción ActivePage indica cuál será la pestaña que se verá iniciando la sesión del usuario. Con esto se cumple con los requerimientos de cada tipo de usuario, localizando el nivel de privilegios que cada uno posee buscando por nombre en la tabla USUARIO. Cada pestaña se numera del 1 al 5 de derecha a izquierda siendo:

1. Actualizar Actividad
2. Nueva O.T.
3. Consulta de Producción
4. Edición de Reportes
5. Registro de Usuarios

En la siguiente imagen vemos la sesión de Recursos Humanos, aunque seleccione otras pestañas sólo podrá ver la sección que le corresponde:



## 4.2 Impacto en la empresa

Aunque en un principio se creía que utilizar un lenguaje de programación como Visual FoxPro contaría con varias desventajas al ser un lenguaje en desuso y sin actualizaciones por parte de Microsoft fue lo suficientemente efectivo para cumplir con el propósito principal: recabar información de manera digital, procesarla, guardarla y consultarla.

Cuando una compañía mediana que se enfoca en obtener los resultados en el rubro donde se desenvuelve (en este caso impresión) se convierte en su máxima prioridad, al ver que el rendimiento no es suficiente, hay estancamiento y voltea a ver los avances de la competencia, comienzan con un análisis de lo que están haciendo ahora, localizan las áreas de oportunidad y al optimizar los resultados tarde que temprano ven que es necesario invertir en los recursos informáticos.

Cuando se comenzó a desarrollar este producto las expectativas del cliente eran inciertas, gran parte de los requerimientos funcionales del cliente fueron llenados por el desarrollador después de entrevistar a los jefes y supervisores de las diferentes áreas de trabajo.

La información casi ambigua con la que transmitían su necesidad llevó a hacer un listado de objetivos generales que tenía que cumplir SiRPAG y de allí dividir cada objetivo en los requerimientos del cliente y desarrollador, teniendo presente las restricciones del software y hardware que dejaron en claro desde el comienzo.

La figura 3.2 fue la primera parte que se construyó y mostró al cliente antes de continuar con el desarrollo de la interfaz. La expresión del cliente “quiero que la aplicación se parezca a una que vi hace tiempo” hicieron que como desarrolladora tuviera que investigar y obtener los recursos necesarios para cumplir con las expectativas de un cliente que deseaba una aplicación con amigable apariencia en un motor de desarrollo discontinuado. Por esto la colecta de los requerimientos fue de los procesos más tardados en concluir, el cliente no definía por completo lo que necesitaba. Algo importante a la hora de concretar los requerimientos fue la ayuda visual, con una serie de dibujos en una pizarra se permitió acordar la ubicación de gran parte de los elementos de la interfaz.

La empresa comprobó que, por mínimo que se desarrolle un sistema, sea pequeño o grande, desde el planteamiento y toma de requerimientos, los involucrados podrán tener una perspectiva amplia del producto final que se les va a entregar y si servirá para todo lo que necesitan y cumplirá con los objetivos principales por los que fue creado.

La información que se había recolectado hasta hace poco a través del papel se había mostrado insuficiente, la gerencia había notado que los tiempos que daban en el pasado los encargados y lo que ya estaban recolectando con los reportes escritos eran muy opuestos.

SiRPAG dio la solución a la problemática principal, al fin se dieron a conocer cifras fieles a la realidad sumando tiempos imprevistos, reparaciones, preparación del equipo, etc. La información se descarga con las horas y minutos de producción calculadas, se entregan a tiempo las estadísticas a dirección y se trabaja a tiempo los planes de contingencia de los próximos meses.

Tres años atrás de esta publicación la compañía tomó la decisión de buscar un método para obtener el tiempo de la producción de manera más confiable. Anteriormente se consultaba al jefe de área sobre los tiempos, a su vez él y su personal lo platicaban y tentativamente daban una respuesta, así el vendedor tenía que hacer el presupuesto y el cobro al cliente, pero en la realidad se iban a tiempos tardíos y comenzaban los problemas con la falta de cumplimiento al cliente y pérdidas económicas. En el peor de los casos se podía llegar a extraviar la solicitud impresa en producción y nadie se enteraba hasta que el cliente reclamaba su producto o el vendedor consultaba el estatus.

Se optó por una hoja de papel al día por operador, donde redactarían lo que hacían en su día a día en el trabajo, como el de la figura 4.1, pero se presenciaban los siguientes problemas: letra ilegible, datos erróneos o incongruentes con producciones anteriores (no empataban), pérdida de reportes y poco compromiso del trabajador.

FORMATO							NO. DE FORMATO			
REPORTES DIARIOS DE PRODUCCION							PR-001			
							AREA:		T-V	
FECHA	Vie 040817			MAQUINA		Rico 81205				
HORA	CLAVE	OPERADOR	O.T.	CLIENTE	APLICACION	FOL. INICIAL	FOL. FINAL	CANTIDAD	COMENTARIOS	
7:00	2	Sergio Gonzalez	S/n	Cotopris 2016	Listado	7557265			1 Cambio	
7:15	2	/	/	Bayer	/	/	/		99	
7:30	4	/	/	/	P2 Paralel.	/	/		P21	
7:45	4	/	/	/	/	/	/		P2 - 7499	
8:00	4	/	/	/	/	/	/		Fin P2	
8:15	4	/	/	/	/	/	/		A-14 P1	
8:30	4	/	/	/	/	7562267	/		A14 P2 Fin	
8:45	4	/	/	/	/	/	/		FP2	
9:00	4	/	/	/	/	/	/		Fin P3	
9:15	4	/	/	/	/	7567061	/		- Recus. - 200899	
9:30	4	/	/	/	/	/	/		2016 P1	
9:45	4	/	/	/	/	/	/		FP1	
10:00	4	/	/	/	/	7572065	/		506	
10:15	4	/	/	/	/	/	/		- Recus. - 200899	
10:30	4	/	/	/	/	/	/		2016 P1	
10:45	4	/	/	/	/	7577065	/		FP1	
11:00	4	/	/	/	/	/	/		FP1	
11:15	4	/	/	/	/	/	/		FP1	
11:30	4	/	/	/	/	7582065	24800		FP1	
11:45	4	/	/	ATT	Cartas	7584098	7584121	0.023	FP1	
12:00	4	/	/	Bayer	Listado	7584098	7584121	0.023	FP1	
12:15	4	/	/	/	Listado	7584122	/		FP1	
12:30	4	/	/	/	/	/	/		FP1	
12:45	4	/	/	/	/	7589121	/		FP1	
1:00	4	/	/	/	/	/	/		FP1	
1:15	4	/	/	/	/	/	/		FP1	
1:30	4	/	/	/	/	7594121	/		FP1	
1:45	4	/	/	Motiva	Cartas	7594121	7594121		FP1	
2:00	13	Comida		/	/	/	/		FP1	
2:15	13	Comida		/	/	/	/		FP1	
2:30	13	Comida		/	/	/	/		FP1	
2:45	13	Comida		/	/	/	/		FP1	
3:00	13	Comida		/	/	/	/		FP1	
3:15	4	Sergio Gonzalez	S/n	Bayer	Listado	/	/		FP1	
3:30	4	/	/	Falta papel	/	/	/		FP1	
3:45	4	/	/	Retorno	/	/	/		FP1	
4:00	3	/	2203	Motiva	/	7607162	23.039		FP1	
4:15	4	/	/	/	/	/	/		FP1	
4:30	4	/	/	/	/	/	/		FP1	
4:45	4	/	/	/	/	/	/		FP1	
5:00	4	/	/	/	/	/	/		FP1	
5:15	4	T. X T. emp		/	/	/	/		FP1	
5:30	4	/	/	/	/	/	/		FP1	
5:45	4	/	/	/	/	/	/		FP1	
6:00	4	/	/	/	/	/	/		FP1	
6:15	4	/	/	/	/	/	/		FP1	
6:30	4	/	/	/	/	762065	/		FP1	
6:45	4	/	/	/	/	/	/		FP1	
7:00	4	/	/	/	/	7624517	14503		FP1	

Figura 4.1 Ejemplo del reporte escrito en la organización.

Dos empleados encargados del área de Calidad tenían la ardua tarea de capturar la información en computadora, inicialmente los supervisores recolectarían los reportes y se los entregarían capturados, pero los reportes que se recolectaban en una semana de trabajo de los operadores de una sola área de Trabajo, ejemplo, el área de Impresión variable compuesta de 8 personas que sumaban un total de 40 reportes en una semana se los entregaban de 2 a 4 semanas de retraso al área de Calidad.

Cada área tiene diferente cantidad de personal a su disposición, al final debido a que los supervisores y subgerentes se encontraban ocupados con el resto de sus deberes se les dejó la tarea de recolectar los escritos y pasarlos a la computadora únicamente a dos personas del área de Calidad.

Continuando con el ejemplo del área de Impresión Variable, si algún operador redactaba de manera incorrecta su reporte o era poco entendible, a Calidad le llegaba a tomar hasta una hora hacer las aclaraciones con el trabajador debido a que este último llegaba a olvidar lo que había hecho ese día. En el mejor de los casos a los operadores que redactaban de manera correcta su reporte a Calidad les tomaba de 10 a 15 minutos pasarlo a la computadora.

En los primeros meses a Calidad le tomaba 5 días en intervalos de 2 a 3 horas sumando un total hasta de máximo 15 horas semanales revisar, contar y vaciar los reportes en una hoja de cálculo de una semana de trabajo en una sola área. Una semana entera para recolectar lo que había pasado la semana anterior, y aún con la práctica y la mejora de la redacción de los reportes le tomaba a Calidad en 2 semanas en el mismo intervalo de horas recolectar los reportes de un mes de trabajo de sólo 2 áreas operativas.

A pesar de los esfuerzos del área de Calidad para tener a tiempo la información, sólo habían logrado mejorar un poco la recolecta de la información, las 2 áreas operativas sólo pertenecían a una tercera parte de la empresa, Calidad se había convertido en un área de colecta de información y no tenía el tiempo para hacer su trabajo correspondiente como área.

Después de implementar SiRPAG en todas las áreas de trabajo, la información que requerían la descargaban en unos segundos y con la ayuda de una plantilla en Excel lograban preparar las estadísticas de los resultados de producción de todo un mes de trabajo en cuestión de minutos.

El mantener ocupada a un área como es Calidad en capturar reportes escritos gran parte del mes causaba frustración y falta de crecimiento en sus respectivas tareas reales como análisis de pruebas de calidad del producto, validación del visto bueno del cliente y muestra real de producción, medidas para optimizar procesos, etc.

Explicando la figura 4.2 si todos los reportes de un mes se recolectaban y pasaban a la computadora de tan sólo una tercera parte de los operadores (20 trabajadores aproximadamente) en un tiempo de 2 a 3 horas en 10 días hábiles, eso sumaba un total de 20 a 30 horas para recolectar todo el mes de trabajo, si lo multiplicamos por 3 para obtener los resultados de toda la planta suman de 60 a 90 horas de trabajo de recolección, aclaración y captura de la información. Con SiRPAG la información se descarga casi al instante y se obtiene la información de todo el mes de producción de la planta.

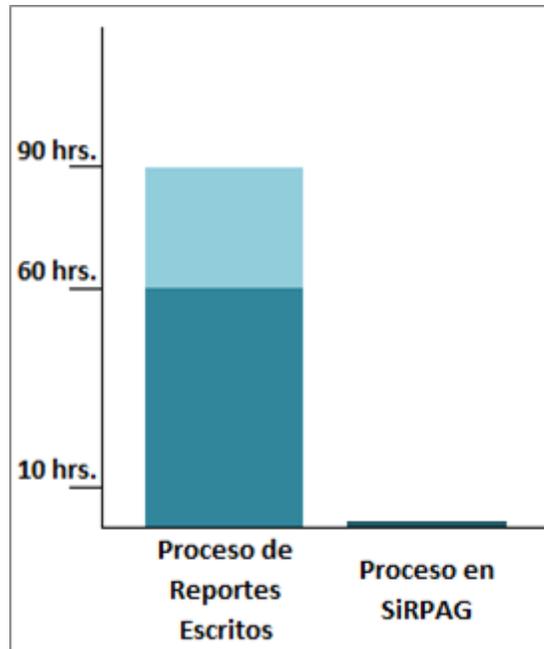


Figura 4.2 Impacto de SiRPAG.

Datos más fidedignos, debido a que en el pasado no contaban con el tiempo que tomaba al operador hacer los paros para preparar la máquina, colocar el material y mantenimiento del equipo de producción, SiRPAG recolectó esta información y mostró el rendimiento del día a día de cada operador.

Permitió a las áreas conocer si por ejemplo un operador tenía un rendimiento de 7,000 ejemplares por hora de producción y al otro mes bajaba a 4,000 ejemplares, se podía saber con quién acercarse y buscar la solución directamente.

También permitió crear conciencia en el operador de que su trabajo estaba siendo evaluado y que sería más difícil falsificar los resultados o solicitar tiempo extra si ya se estaba logrando una mejor planificación de los tiempos gracias al conocimiento real de producción por equipo, ayudando a los que si desempeñaban mejor su trabajo a tener la información de su trabajo y solicitar una compensación por su arduo esfuerzo.

Para algunos operadores fue fácil habituarse debido a que en sus anteriores trabajos ya existían sistemas de captura similares. Toda la información está facilitando realizar cotizaciones más próximas a la realidad y crean objetivos alcanzables: subir producción al mes, recompensas a los que realmente están trabajando y los operadores ya tienen un historial de rendimiento propio con el cual conocer sus áreas de oportunidad.

Los usuarios están mostrando una confianza grande en el sistema, verifican su productividad semana a semana y tienen la seguridad de que el trabajo realizado está siendo evaluado de forma imparcial y objetiva motivándoles a cumplir metas con recompensas a corto y largo plazo.

### 4.3 Conclusiones

Leyendo detenidamente el impacto y resultados se puede declarar algo que era una realidad: no existía una metodología implementada en el desarrollo de sistemas de software en la empresa.

En un principio sólo se recibía un correo de parte del vendedor, supervisor, gerente o director de la empresa con una lista de lo que querían que se hiciera, se desarrollaba por parte del programador, se entregaba y comenzaban a quitar lo que no necesitaban y poner nuevos elementos que se requerían, se hacían algunas pruebas de caja negra y se entregaban los resultados. Si el sistema requería cambios el programador que lo desarrolló lo parcheaba y si se corría con suerte dejaba un respaldo del proyecto anterior en alguna carpeta de su equipo sin ninguna documentación, ni el código tenía comentarios en el peor de los casos.

Esta manera de trabajar era una versión incompleta de la metodología de prototipo pero que evidentemente no funcionaba. La metodología en cascada modificada que utilicé para este desarrollo le parecía algo rígida para el cliente al comienzo, un listado formal de requerimientos y alcance del cliente y del desarrollador, los diagramas de diseño, interfaz y apariencia, desarrollo, pruebas y muestras de visto bueno confirmadas por el cliente, entrega del producto y documentación operativa y técnica. Toda esta formalidad para la directiva era “demasiado para algo sencillo y para una empresa no tan grande”, pero viendo el resultado se llegó a la conclusión:

- El tiempo que se invirtió al definir los requerimientos permitió que el desarrollo fuera fluido sin tener que volver a molestar al cliente con preguntas y que el desarrollador se detuviera a investigar. No se invertiría tiempo en correcciones grandes ya que no existiría reprogramación para ocultar los errores o malentendidos en el desarrollo.
- El sistema tenía un objetivo real que cumplía toda una necesidad de la compañía. Existía al fin una comunicación con Mecano que retroalimentaba a SiRPAG para ofrecer una solución completa la consulta y análisis de información.
- Si desarrollaban sus productos con la metodología en cascada modificada al fin contarían con un calendario real que podían compartir con el cliente y definir la fecha de cierre sin retrasos alarmantes. Hace tiempo se había intentado hacer un sistema que apoyara a la empresa con la colecta de la información, pero no se había logrado concretar ya que no se tomaba el tiempo los responsables en juntar los requerimientos y el desarrollador no seguía una metodología y no comprendía el alcance que tenía que tener el sistema.
- Eso se podía transportar al desarrollo de las pequeñas aplicaciones que generarían de manera automática los productos frecuentes de los clientes que contaba la empresa. Y si faltaba el desarrollador encargado de alguna de las aplicaciones no existiría el inconveniente de liberar el producto ya que se tendría una documentación operativa y entendimiento de lo que hace el programa con los modelos de proceso y estados.
- Los resultados se comenzaron a ver desde los primeros meses obteniendo al fin información con datos fidedignos que la sustenten. Gracias al entorno amigable de SiRPAG los empleados se familiarizaron rápidamente con el sistema.

- No sólo fue un beneficio para la empresa, sino cada empleado al fin tendría la manera de consultar su bitácora de trabajo y definirse metas.
- Sin cambios drásticos por hacer gracias a que la metodología orilla a que se concluya una fase para poder pasar a la siguiente, evitamos que el desarrollador tire trabajo invertido.
- Se crea una cultura de organización y definición de los objetivos de una forma más detallada y sólida.
- Se disminuye gran parte la mala costumbre de desarrollar sin un plan de trabajo permitiendo tiempo para desempeñar en nuevos proyectos con la seguridad de que lo que se desarrolló cumple por completo con su tarea.

Desde que la metodología fue propuesta hace casi 50 años se pensaría que con la creación de nuevas técnicas de desarrollo ésta quedaría en el olvido, pero se demuestra que la forma rigurosa que implanta la metodología de trabajar por etapas y no avanzar hasta que se concluyó el bloque anterior permite progresar de manera fluida y sin juntas innecesarias para aclarar dudas cuando el sistema tenga ya un desarrollo considerable y estar volviendo atrás.

Esta metodología aprendida en la carrera, más los conocimientos de planeación, desarrollo y programación de aplicaciones, me permitieron dar frente a la demanda de una Empresa Mexicana en crecimiento que ha quedado satisfecha con los resultados que ya son palpables en todas sus áreas operativas.

Sin afán de comparar el trabajo de los desarrolladores del nuevo Sistema de Cotización y Generación de Pedidos Mecano que se encuentran desarrollando por módulos y prototipos, se le demostró a la empresa de Excelform lo importante que es delimitar el proyecto y establecer los requerimientos desde el comienzo y así evitar retrasos y reprocesos que han costado miles a la Compañía.

Cuando se realizaron las entrevistas correspondientes para recabar la información de los requerimientos, no todos los empleados y/o supervisores mostraban empatía ante la situación, se mostraban a veces hasta herméticos en proporcionar datos, no estaban familiarizados con la metodología, pero al final los resultados hablaron por sí mismos.

En la actualidad cuando lo que se desea es que las aplicaciones se tengan a tiempo y listas para cumplir su propósito, es importante construir sus cimientos de forma certera y detallada para ser una estructura firme ante los errores que “los parches” provocan a causa de los reprocesos y cambios de última hora, con el uso de una metodología tan estructurada como lo es la metodología en castada modificada se obtuvo un sistema completo que satisface las necesidades del cliente y no un frankenstein a la merced de fracasar a causa de su desarrollo poco confiable.

## Glosario de términos

---

- **ALTER**

Sentencia en SQL que permite modificar la estructura de un objeto, por ejemplo, ALTER TABLE permite agregar, modificar o borrar columnas.

- **Cantidad Escalar**

Define una magnitud en determinada unidad. Ejemplo: se realizó una producción de 20 piezas, siendo 20 la magnitud y piezas el tipo de unidad.

- **Checkbox**

Componente de interacción presente en la interfaz gráfica de usuario para seleccionar una condición o un conjunto de opciones. Se compone comúnmente de un recuadro o círculo que al ser seleccionado se llena su contenido con un símbolo (sea un punto, tache, etc.).

- **CMD**

Su nombre es símbolo del sistema o command prompt y es el intérprete de comandos del sistema operativo Windows.

- **CREATE**

Sentencia en SQL para crear un nuevo objeto, el tipo de objeto se determina con la palabra después de CREATE, por ejemplo, CREATE DATABASE es para crear una nueva base de datos.

- **CURSOR**

Elemento que contiene datos determinados de una consulta en SQL, en el cursor se puede recorrer fila a fila y modificar resultados sin comprometer el contenido de ninguna tabla.

- **Dato**

Es una representación simbólica de algún tipo de descripción, ya sea la edad de una persona, número de factura, cuenta bancaria, fecha de nacimiento, etc.

- **DELETE**

Sentencia en SQL que marca los registros que serán eliminados. No elimina de manera física hasta usar la instrucción PACK y pueden ser recuperados con RECALL.

- **DLL**

Su nombre es *dynamic-link library* traducido al español como biblioteca de enlace dinámico término que se refiere a archivos que contienen código ejecutable cargados bajo la demanda de un programa en el sistema operativo.

- **DROP**

Sentencia en SQL que permite borrar un objeto, por ejemplo, DROP DATABASE elimina una base de datos por completo y la única manera de restaurarla es a través de una copia de seguridad.

- **Estándar**

Son un conjunto de normas que permiten comprobar si un producto o proceso está cumpliendo con las características obligadas que benefician al usuario final.

- **Información**

Conjunto de datos relacionados que a través de un proceso permite tomar decisiones dependiendo de su contexto y propósito.

- **INSERT**

Sentencia en SQL que inserta un registro al final de la tabla en la columna especificada.

- **Módulo de Control**

Se refiere al módulo principal en los diagramas de navegación de la interfaz en el diseño de un sistema. Abreviado como MC, es el nombre a la ejecución de un sistema.

- **ODBC**

(Open DataBase Connectivity) Es un estándar de acceso a bases de datos a través de una interfaz de programación de aplicaciones (API). A través del uso de sus sentencias se puede acceder a diversos tipos de archivos de bases de datos, ej. MySQL, FoxPro, Excel, dBase, Texto, DB2, etc.

- **OLE DB**

(Object Linking and Embedding for Databases) En español enlace e incrustación de objetos para bases de datos es una interfaz de distribución de datos de Microsoft, un estándar abierto que permite el acceso universal de los datos.

- **ORDER BY**

Sentencia en SQL que ordena los datos de manera ascendente o descendente de una o más columnas de una consulta.

- **Textfield**

Componente presente en cualquier interfaz de usuario, se traduce como “caja de texto” y su función es recibir elementos ya sean numéricos y/o alfabéticos, etc. También se pueden hacer cajas de texto no editables para únicamente presentar algún tipo de información textual.

- **TRUNCATE**

Sentencia en SQL que elimina todos los datos de una tabla conservando su estructura.

- **UPDATE**

Sentencia en SQL que actualiza el contenido de una tabla. A través de WHERE se elige cuáles serán los registros que obtendrán nuevos valores, si omitimos esta sentencia realizará el cambio en toda la tabla.

## Bibliografía

---

- **Alonso, F., Martínez, L. y Segovia, F. J.** (2005). *Introducción a la Ingeniería del Software: Modelo de Desarrollo de Programas*. Primera Edición. Madrid, España: DELTA.
- **Calidad del Software.** (2014). [Blog] *Jummp Gestión de proyectos y desarrollo de software*. Recuperado de <https://jummp.wordpress.com/category/calidad-del-software-2/> [Accedido el 20 Abr. 2017].
- **CCM.** (2017). *Visual FoxPro*. [online] Recuperado de <http://es.ccm.net/faq/3746-visual-foxpro#top> [Accedido el 18 Feb. 2017].
- **Ceballos Cardona, Y. y Velez Tascon, L.** (2013). *Norma ISO 12207 Information Technology/ Software Life Cycle Processes*. [PDF] Colombia. Recuperado de <https://es.scribd.com/doc/131847881/NORMA-ISO-12207-pdf> [Accedido el 20 Ene. 2017].
- **CodePlex.** (2017). *VFPX*. [online] Recuperado de <https://vfp.codeplex.com/> [Accedido el 15 Ene. 2017].
- **Cortes, L.** (2017). *6.2 Modelo de Casos de Uso*. [online] 148.204.211.134. Recuperado de [http://148.204.211.134/polilibros/Portal/Polilibros/P\\_proceso/ANALISIS\\_Y\\_DISEÑO\\_DE\\_SISTEMAS/IngenieriaDeSoftware/CIS/UNIDAD%20VI/6.2.htm](http://148.204.211.134/polilibros/Portal/Polilibros/P_proceso/ANALISIS_Y_DISEÑO_DE_SISTEMAS/IngenieriaDeSoftware/CIS/UNIDAD%20VI/6.2.htm) [Accedido el 21 Feb. 2017].
- **Ecured.cu.** (2017). *Estándares para el mantenimiento del software – EcuRed*. [online] Recuperado de [https://www.ecured.cu/Est%C3%A1ndares\\_para\\_el\\_mantenimiento\\_del\\_software](https://www.ecured.cu/Est%C3%A1ndares_para_el_mantenimiento_del_software) [Accedido el 20 Abr. 2017].
- **Hernández Briones, A.** (2017). *Estándares Biométricos*. [online] Redyseguridad.fi-p.unam.mx. Recuperado de <http://redyseguridad.fi-p.unam.mx/proyectos/biometria/estandares/estandar.html> [Accedido el 10 Mar. 2017].
- **López Gaona, A.** (Sin Fecha). *El modelo Entidad-Relación*. [PDF] Ciudad de México. Recuperado de <http://hp.fcencias.unam.mx/~alg/bd/er.pdf> [Accedido el 23 Mar. 2017].
- **López Lira, F.** (2007). El estándar IEEE 1074-1997. [Blog] *Ingeniería de Software*. Recuperado de <http://issegunfl.blogspot.mx/2007/04/el-estndar-ieee-1074-1997.html> [Accedido el 18 May. 2017]
- **Martín Quetglás, G. & Morillo Tena, P.** (2002). *Ingeniería Informática. Tema 9: Mantenimiento* [PDF] Valencia, España. Recuperado de <http://informatica.uv.es/iiguia/2000/IPI/material.htm> [Accedido el 17 Abr. 2017].
- **Miñana, R.** (2012). AOS2012 Gestión del código heredado. [Blog] *Calidad y Software*. Recuperado de <https://calidadyssoftware.wordpress.com/2012/06/26/aos2012-gestion-del-codigo-heredado/> [Accedido el 17 Abr. 2017].
- **Msdn.microsoft.com.** (2017). *Visual FoxPro Home*. [online] Recuperado de <https://msdn.microsoft.com/en-us/library/mt490117> [Accedido el 13 Feb. 2017].

- **Piattini, M., García, F. y Caballero, I.** (2007). *Calidad de Sistemas Informáticos*. Primera Edición. Alfaomega Grupo Editor.
- **Pressman, R.** (2010). *Ingeniería de Software. Un enfoque práctico*. Séptima Edición. España: McGraw-Hill.
- **Silberschatz, A., Korth, H. y Sudarshan, S.** (2006). *Fundamentos de Bases de Datos*. Quinta Edición. España: McGraw-Hill.
- **Sommerville, I.** (2011). *Ingeniería de Software*. Novena Edición. Madrid, España: PEARSON EDUCACIÓN.
- **Torres, M.** (2017). *Fundamentos del diseño de software*. [PDF] Almería, España. Recuperado de <http://indalog.ual.es/mtorres/LP/FundamentosDiseno.pdf> [Accedido el 5 Mar. 2017]
- **Msdn.microsoft.com.** (2017). *Modelar la Arquitectura de la Aplicación*. [online] Recuperado de <https://msdn.microsoft.com/en-us/library/dd490886.aspx> [Accedido el 24 Jun. 2017].
- **Dlsi.ua.** (2008). *Diagrama de Flujo de Datos – Jaime*. [PDF] Recuperado de [ftp://www.dlsi.ua.es/people/jaime/apuntes/aesi\\_cap4.pdf](ftp://www.dlsi.ua.es/people/jaime/apuntes/aesi_cap4.pdf) [Accedido el 20 Jul. 2017].
- **Sparxsystems.com.ar.** (2007). *Diagrama de Máquina de Estado*. [online] Recuperado de [http://www.sparxsystems.com.ar/resources/tutorial/uml2\\_statediagram.html](http://www.sparxsystems.com.ar/resources/tutorial/uml2_statediagram.html) [Accedido el 04 May. 2017].
- **Zaldívar Z., Orlando** (2011). *Apuntes de la asignatura de Ingeniería de Software*. [book][Consultado el 30 Jul. 2017].