

# Universidad de las Ciencias Informáticas



## Módulo de Supervisión y Control

---

del Sistema Único de Identificación Nacional

### *Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas*

#### **Autores:**

Yadira Castillo Medinilla

René Abilio Piñeiro Hernández

#### **Tutor(es):**

Msc. Yudenia Ramírez Mastrapa

Ing. Gleydis Claro Morgado

Junio de 2010

*Si supiese que es lo que estoy haciendo, no lo llamaría  
investigación, ¿verdad?*

*Albert Einstein*

## *Declaración de autoría*

---

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los \_\_\_ días del mes de \_\_\_ del año\_\_\_.

\_\_\_\_\_

Yadira Castillo Medinilla

**Firma del Autor**

\_\_\_\_\_

René Abilio Piñeiro Hernández

**Firma del Autor**

\_\_\_\_\_

Msc. Yudenia Ramírez Mastrapa

**Firma del Tutor**

\_\_\_\_\_

Ing. Gleydis Claro Morgado

**Firma del Tutor**

## *Datos de contacto*

---

**Tutora:** Msc. Yudenia Ramírez Mastrapa

- Ingeniera Informática, CUJAE, 2003
- Categoría docente: Profesora Asistente.
- Máster en Gestión de Proyectos Informáticos.
- Profesora del Departamento de Sistemas Digitales, Facultad 1, Universidad de las Ciencias Informáticas
- Jefa de departamento de Soluciones de software del Centro de Identificación y Seguridad Digital.

**Correo electrónico:** yudenia@uci.cu

**Tutora:** Ing. Gleydis Claro Morgado

- Ingeniera Informática, UCI, 2009.
- Categoría docente: Adiestrada
- Trabaja en el Departamento de Automatización de la Dirección de Identificación y Registro del Ministerio del Interior.
- Analista de sistema del Sistema de Identificación, Migración y Extranjería.

**Correo electrónico:** gclaro09@graduados.uci.cu

*A la obra infinita que es la Revolución Cubana...*

*A mis abuelos, por haber sido los primeros en darme amor...*

*A mis padres, por el cariño infinito...*

*A mis hermanos y amigos...*

*A la vida, por esta oportunidad...*

**Yadira.**

*A mi familia.*

*A mi novia.*

*A mis amigos de ahora.*

*A mis amigos de antes.*

*A todos los que declararon una sola variable en aras de lograr este dulce sueño.*

*A todos: Gracias de corazón.*

**René**

A la Revolución Cubana y al Comandante Fidel por la oportunidad de una formación profesional única.

A mis madres Beatriz y Julia, por ser padres y madres en los momentos más difíciles, por estar siempre en cada paso que doy, por dejarme ser su “niña”.

A mi papi Víctor, por permitirme conocer el cariño de un verdadero padre, por apoyarme y querer hacerme siempre feliz.

A mis hermanitas Yairima y Yahima por el compromiso de ser un ejemplo para ellas, a mis hermanos Alier, Vitico, Alexander y Damián por haberme acogido en la familia, por cuidarme siempre y enseñarme un poco de la vida.

A mi enorme familia de muchas mujeres y pocos hombres, a todas esas Castillo que se preocupan por mí, a mis primas, primos y tíos.

A Yude y Gleydis, porque criticaron este documento sin miserias para que resultara útil.

A René, por ser el mejor de los compañeros de tesis que ha existido, por no dejarme nunca sola y aguantar el diseño de este trabajo sin protestar.

A los Jefes que nunca nos dejaron solos batallando con el workflow, a Blanco y Humberto.

A Ernesto, por ser el mejor de los amigos en mucho tiempo.

A mis amigos de los años y a los de ahora, a May, a Nany, a Danep, Irene, Mayitin, Shortby, Impy, Iro, Yune y Malena. Gracias por tolerarme con mis defectos, por quererme siempre.

A los amigos que tienen SUIN hasta para regalar. A Dianita (“Venérea”), por estos meses súper divertidos, a Ale (“Fefa”), por haberme enseñado tanto en tan poco tiempo, a Ely (“Rita”) por los sabios consejos, a Pacheco (“Checo”) por querer siempre lo mejor para mí, a Valdés (“Crixus”) por hacerme reír muchas veces, a Anita (“Caridad”) por escucharme y a Lin (“La Pelua”) que a pesar de ser una juez imparcial siempre estuvo de mi lado.

A todos los residentes de 202 que formaron una familia digna de película.

A los muchachos del Estado Mayor porque aprendí mucho con ellos y viví momentos muy lindos formando ese sueño.

*A los amigos de la familia en especial a Nersa, China, Ivón, Michel y Vladimir, por quererme mucho y considerarme ejemplo.*

*A todas las personas que he tenido el placer de conocer y de las cuales me llevo un poco de ternura. A todos los que gozan de mi amistad y me consideran su compañera.*

### **Yadira**

*A mi madre y mi padre, por hacer de mí un “tipo” tan completo.*

*A mi novia, por soportar, querer, amar, luchar, consentir y malcriar.*

*A la memoria de mi tío Abilio, bujía de cada paso que doy, te lo dedico donde quiera que vivas hoy.*

*A mis dos hermanos, Miguel y Carlo Carlo, hermanos de sangre y corazón.*

*A toda mi familia, desde Laurencita hasta mi primo Troa.*

*A la Yadira, por luchar juntos esta última batalla, por ser como es, por sus masajes y por saber tanto Word.*

*A mis dos tutoras, Yudenia y Gleydis, por criticar y a la vez correr con nuestros gritos.*

*A la familia que gané y afiancé durante estos 5 años, a Dayron, a Chucho, a Sam, a Yunnier y a Alexis.*

*A la profe Yadira, o mejor, “A mi amiga Yadira”.*

*A David y Jorge Dayro, que no hicieron nada, pero igual quiero agradecerles.*

*Al SUIN de 202, a los Alejandro's, a la Diana, a los Jefes, por guiar esta locura con cordura.*

*A todos mis amigos y mis enemigos.*

*A mis Industriales.*

*Gracias.*

### **René**

El auge y desarrollo de las Tecnologías de Informática y Comunicaciones (TICs) a nivel mundial ha traído consigo la revolución de muchos procesos que hasta el momento requerían de tiempo y organización en demasía. En ese rápido avance se encuentran los sistemas de identificación de personas que, si bien ayudan a controlar las identidades de los ciudadanos, garantizan un control casi total sobre los procesos identificativos.

El Ministerio del Interior de la República de Cuba, como parte de su trayecto hacia la informatización, se ha propuesto la creación de un software para la Dirección de Identificación y Registro (DIR) que garantice rapidez, eficiencia y gran capacidad de manejo de datos; de forma que permita controlar a cada ciudadano de este país en un único sistema. Para el desarrollo del mismo se han estudiado las metodologías, procesos y estrategias adecuadas que permitan centralizar los procesos de identificación.

En el presente trabajo de diploma se presenta un Módulo de Supervisión y Control, perteneciente al Sistema Único de Identificación Nacional, que tiene como objetivo desarrollar una solución de software para la supervisión y control de trámites de obtención de documentos de identidad. De esta forma se logran simplificar los errores en los procesos y se garantiza claridad, confiabilidad, y legitimidad a dichos documentos.

**Palabras claves:** metodologías, procesos, software, documentos de identidad.



<b>INTRODUCCIÓN .....</b>	<b>12</b>
<b>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....</b>	<b>16</b>
INTRODUCCIÓN .....	16
1.1 ANTECEDENTES DE LOS PROCESOS DE SUPERVISIÓN Y CONTROL EN LAS OFICINAS DE CIRP.....	16
1.2 CONCEPTOS ASOCIADOS AL DOMINIO DEL PROBLEMA .....	16
1.3 ANÁLISIS DE OTRAS SOLUCIONES EXISTENTES .....	17
1.3.1 <i>Servicio Administrativo de Identificación Inmigración y Extranjería.....</i>	17
1.3.2 <i>EMIPAS: Sistema Cubano de Emisión de Pasaportes.....</i>	19
1.3.3 <i>Sistema de Gestión Académica Akademos .....</i>	20
1.3.3.1 Inconvenientes de los sistemas analizados.....	20
1.4 AMBIENTE DE DESARROLLO .....	21
1.4.1 <i>Desarrollo basado en procesos.....</i>	22
1.4.2 <i>Metodología de desarrollo.....</i>	22
1.4.3 <i>Notación para el Modelado de Procesos de Negocio .....</i>	23
1.4.4 <i>Herramienta para la gestión de Base de Datos.....</i>	24
1.4.5 <i>Herramientas para el entorno de desarrollo integrado .....</i>	25
1.4.6 <i>Herramienta para el modelado de datos.....</i>	30
1.4.7 <i>Herramienta para la obtención de reportes.....</i>	30
CONCLUSIONES .....	32
<b>CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....</b>	<b>33</b>
INTRODUCCIÓN .....	33
2.1 DESCRIPCIÓN DEL PROCESO DE SUPERVISIÓN ACTUAL .....	33
2.2 DIAGRAMA DEL PROCESO DE SUPERVISIÓN.....	33
2.3 ESPECIFICACIÓN DE REQUISITOS FUNCIONALES .....	36
2.3.1 <i>Concepción de los modelos de mejora de procesos.....</i>	36
2.3.2 <i>Roles del sistema.....</i>	41
2.3.3 <i>Catálogo de requisitos funcionales.....</i>	41
2.3.4 <i>Descripción de requisitos funcionales.....</i>	43
2.4 MODELO DE ENTIDADES CONCEPTUALES.....	45
2.4.1 <i>Descripción de las entidades fundamentales.....</i>	47
2.5 ESPECIFICACIÓN DE REQUISITOS NO FUNCIONALES .....	47
<i>Usabilidad .....</i>	47
<i>Seguridad .....</i>	48
<i>Interfaz.....</i>	49
Interfaces de usuario .....	49
Interfaces Hardware.....	50
Interfaces Software.....	50
Interfaces de Comunicación.....	50
CONCLUSIONES .....	51
<b>CAPÍTULO 3: ANÁLISIS Y DISEÑO.....</b>	<b>52</b>
INTRODUCCIÓN .....	52
3.1 ARQUITECTURA DE LA SOLUCIÓN .....	52

3.1.1	<i>Capas de los módulos de la aplicación</i> .....	52
3.2	PATRONES DE DISEÑO.....	56
3.2.1	<i>Patrones</i> .....	56
3.2.2	<i>Patrones Workflow</i> .....	57
3.3	ESPECIFICACIONES DE LAS CLASES.....	57
3.3.1	<i>Clases Controladoras</i> .....	57
3.3.2	<i>Clases Conectoras</i> .....	58
3.3.3	<i>Clases Entidades</i> .....	58
3.3.4	<i>Diagrama de clases del diseño</i> .....	61
3.4	SERVICIOS DEL SISTEMA.....	61
3.4.1	<i>Descripción de los servicios del sistema</i> .....	62
3.5	DISEÑO DE WORKFLOW.....	63
3.6	MODELO DE DATOS.....	68
3.6.1	<i>Descripción de las entidades fundamentales</i> .....	70
3.7	CONCLUSIONES.....	71
<b>CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA</b> .....		<b>72</b>
INTRODUCCIÓN.....		72
4.1	ESTÁNDARES DE CODIFICACIÓN Y TRATAMIENTOS DE ERRORES.....	72
4.1.1	<i>Estilos para la capitalización</i> .....	72
4.1.2	<i>Sensibilidad a mayúsculas</i> .....	73
4.1.3	<i>Tratamiento de errores</i> .....	73
4.2	DIAGRAMA DE COMPONENTES.....	74
4.3	DIAGRAMA DE DESPLIEGUE.....	75
4.4	INTERFACES DEL SISTEMA.....	76
4.5	PRUEBAS.....	78
4.5.1	<i>Diseño de Casos de Prueba</i> .....	79
4.5.2	<i>Pruebas Unitarias</i> .....	79
4.5.3	<i>Pruebas de sistema</i> .....	80
4.5.3.1	Iteraciones de prueba.....	81
4.5.4	<i>Resultado de las pruebas</i> .....	82
4.6	CONCLUSIONES.....	83
<b>CONCLUSIONES GENERALES</b> .....		<b>84</b>
<b>RECOMENDACIONES</b> .....		<b>85</b>
<b>REFERENCIAS BIBLIOGRÁFICAS</b> .....		<b>86</b>
<b>GLOSARIO DE TÉRMINOS</b> .....		<b>89</b>

<b>FIG. 1</b> DIAGRAMA DEL PROCESO CONFECCIÓN .....	34
<b>FIG. 2</b> DIAGRAMA DEL PROCESO ENTREGA .....	35
<b>FIG. 3</b> MACRO PROCESO SUPERVISAR TRÁMITE .....	37
<b>FIG. 4</b> SUBPROCESO SUPERVISAR DATOS .....	38
<b>FIG. 5</b> PROCESO IMPRIMIR DIP .....	39
<b>FIG. 6</b> PROCESO VISUALIZAR EXPEDIENTE DE LA PERSONA .....	40
<b>FIG. 7</b> PIU SUPERVISAR DATOS .....	45
<b>FIG. 8</b> MODELO DE ENTIDADES CONCEPTUALES .....	46
<b>FIG. 9</b> COMPONENTES DE LOS MÓDULOS DE LA APLICACIÓN .....	53
<b>FIG. 10</b> DIAGRAMA DE CLASES DEL DISEÑO DEL REQUISITO FUNCIONAL IMPRIMIR DIP .....	61
<b>FIG. 11</b> DISEÑO DE WORKFLOW PARA EL MÓDULO DE SUPERVISIÓN Y CONTROL .....	68
<b>FIG. 12</b> MODELO DE DATOS DEL MÓDULO DE SUPERVISIÓN Y CONTROL .....	69
<b>FIG. 13</b> DIAGRAMA DE COMPONENTES DEL MÓDULO DE SUPERVISIÓN Y CONTROL .....	74
<b>FIG. 14</b> DIAGRAMA DE DESPLIEGUE DEL MÓDULO DE SUPERVISIÓN Y CONTROL .....	75
<b>FIG. 15</b> INTERFAZ SUPERVISAR DATOS .....	76
<b>FIG. 16</b> INTERFAZ SUPERVISAR IMÁGENES .....	77
<b>FIG. 17</b> INTERFAZ IMPRIMIR DIP .....	78
<b>FIG. 18</b> PRUEBA UNITARIA DEL SERVICIO BUSCAR PROCESO .....	80
<b>FIG. 19</b> RESULTADO DE LAS PRUEBAS DEL SISTEMA .....	82
<b>FIG. 20</b> RESULTADO DE LA PRUEBA UNITARIA REALIZADA AL SERVICIO BUSCAR PROCESO .....	83

<b>TABLA 1</b> ROLES DEL SISTEMA .....	41
<b>TABLA 2</b> DESCRIPCIÓN DE LA ENTIDAD DTRAMITE .....	47
<b>TABLA 3</b> DESCRIPCIÓN DE LA ENTIDAD DPERSONA (DNACIONAL) .....	47
<b>TABLA 4</b> DESCRIPCIÓN DE LA ENTIDAD DDOCUMENTO .....	47
<b>TABLA 5</b> DESCRIPCIÓN DE LA CLASE CONTROLADORA IPERSONCONTROLLER .....	58
<b>TABLA 6</b> DESCRIPCIÓN DE LA CLASE CONECTORA IPERSONCONNECTOR .....	58
<b>TABLA 7</b> DESCRIPCIÓN DE LA CLASE ENTIDAD DOREQUEST .....	59
<b>TABLA 8</b> DESCRIPCIÓN DE LA CLASE ENTIDAD PERSONA .....	60
<b>TABLA 9</b> SERVICIO ICAPTARDATOSDOCRSERVICE .....	62
<b>TABLA 10</b> ICHANGESPERSONSUPERVICERSERVICE .....	63
<b>TABLA 11</b> SERVICIO IPRINTDIPRSERVICE .....	63
<b>TABLA 12</b> SERVICIO ISEARCHPERSONSERVICE .....	63
<b>TABLA 13</b> ENTIDAD PERSONA .....	70
<b>TABLA 14</b> ENTIDAD PERSONA NACIONAL .....	70
<b>TABLA 15</b> ENTIDAD PERSONAOTROS DATOS .....	70
<b>TABLA 16</b> ENTIDAD PERSONAÍMAGEN .....	70
<b>TABLA 17</b> DISEÑO DE CASO DE PRUEBA PARA EL REQUISITO SUPERVISAR DATOS .....	79
<b>TABLA 18</b> RESULTADO DE LA TERCERA ITERACIÓN DEL REQUISITO SUPERVISAR DATOS .....	82

### Introducción

Desde el año 1997, en la República de Cuba, se otorga la gestión y control de los procesos relativos a la identificación de los ciudadanos nacionales y extranjeros que residen en la isla a la Dirección de Identificación y Registro (DIR) del Ministerio del Interior (MININT).

Durante el último período del año 2008 y comienzos de 2009 se ha realizado un estudio del Sistema Nacional de Identificación de la Población en favor de garantizar documentos y procesos seguros, con el fin de obtener un Registro Único de Identificación de la Población.

En la actualidad, gran parte del flujo de trabajo es realizado con un nivel de informatización escaso o de forma manual, lo que ha generado trabas en el correcto funcionamiento del mismo. La no existencia de una política que garantice y haga confiable el intercambio de información entre las distintas organizaciones genera descontrol y distintas salidas para el traslado de datos.

El Ministerio del Interior de la República de Cuba se ha propuesto la creación de un software para la Dirección de Identificación y Registro (DIR) que garantice rapidez, eficiencia y gran capacidad de manejo de datos; de forma que permita controlar a cada ciudadano en un sistema único. Se pretenden diseñar herramientas que controlen la calidad de la información así como medir la eficiencia del trabajo que se realiza en las oficinas por parte de los funcionarios. Para el desarrollo del mismo se han estudiado las metodologías, procesos y estrategias adecuadas que permitan obtener como resultado la centralización de las necesidades, fortaleciendo con dicho propósito las relaciones de trabajo y de colaboración con la Universidad de las Ciencias Informáticas, institución comprometida con la informatización de la sociedad.

Las actividades realizadas dentro de cada oficina de Carné de Identidad y Registro de la Población (CIRP) del país necesitan fiabilidad. En la actualidad los procesos relativos a la supervisión de los trámites no cuentan con la confianza requerida. La pérdida de recursos por fallas en los datos y la poca seguridad en lo referente a los documentos legales que autorizan el trámite generan incongruencias que dan al traste con la organización de los eventos, por lo que se necesita garantizar la calidad de los documentos identificativos que se generan.

A partir de la situación problemática antes expresada se plantea como **Problema Científico** a resolver: *¿Cómo proveer un mecanismo para la supervisión y control de trámites en el Sistema Único de Identificación Nacional de la República de Cuba?*

A partir del problema científico se puede inferir como **Objeto de estudio** para la investigación: *Los procesos de supervisión y control asociados a la tramitación en las oficinas de Carné de Identidad y Registro de la Población (CIRP)*, y como **Campo de acción**: *La supervisión y control de los trámites en la obtención del documento de identidad en las oficinas de CIRP.*

La presente investigación tiene como **Objetivo General**: *Implementar una solución de software que permita la supervisión y control de los trámites de obtención del documento de identidad* y como **Objetivos específicos**:

- ❖ Desarrollar el marco teórico de la investigación.
- ❖ Realizar el modelado del negocio para la solución del software.
- ❖ Obtener requisitos funcionales y no funcionales para la supervisión y control del trámite.
- ❖ Realizar el diseño de la solución de software.
- ❖ Realizar la implementación del software.
- ❖ Garantizar la calidad del software de acuerdo a las especificaciones del cliente.

Para darle cumplimiento a los objetivos generales y específicos, se plantea como **hipótesis**: *“Con la implementación del Módulo de Supervisión y Control del SUIN, se obtendrá una aplicación de software, que facilite la supervisión y control de los procesos asociados a la tramitación en las oficinas de CIRP”.*

De la hipótesis anteriormente planteada se obtienen como variables de la investigación:

- ❖ Variable Independiente: Módulo de Supervisión y Control.
- ❖ Variables dependientes: Supervisar y controlar procesos de tramitación.

Para cumplir con lo antes descrito se han trazado las siguientes **tareas investigativas**:

- ❖ Realización de un estudio de los sistemas existentes de supervisión y control de trámites.
- ❖ Realización de un estudio bibliográfico que permita fundamentar las herramientas informáticas y metodologías utilizadas para desarrollar dicha aplicación.
- ❖ Identificación de los procesos de negocio.
- ❖ Modelación y descripción de los procesos de negocio.
- ❖ Aplicación de técnicas de recopilación de información a los proveedores de requisitos.
- ❖ Especificación de los requisitos funcionales y no funcionales del software
- ❖ Realización de los prototipos de interfaz de usuario.

- ❖ Realización de la validación de los requisitos.
- ❖ Ejecución del diseño del software.
- ❖ Implementación del módulo para la supervisión y control de trámites.
- ❖ Ejecución de las pruebas de aceptación.

Para asistir las tareas de la investigación se emplearán los siguientes **métodos científicos**:

### **Métodos teóricos:**

❖ **Histórico lógico:** En la primera parte de la investigación se desarrollará un estudio del estado del arte de la problemática; así como se analizarán las ventajas y desventajas de cada una de las herramientas utilizadas actualmente por las oficinas de CIRP para la gestión de Supervisión y Control de trámites.

❖ **Analítico – sintético:** Se utilizará para captar y resumir varios documentos y procedimientos legales por los cuales se rige la Dirección de Identificación y Registro, en la elaboración del documento de identidad. De ellos se extraerán las ideas fundamentales y al mismo tiempo se detallará la información necesaria para una correcta fiabilidad en los procesos de negocio.

❖ **Hipotético-Deductivo:** Se utiliza para, a partir de la hipótesis, llegar a nuevos conocimientos y predicciones, los que posteriormente son sometidos a verificaciones empíricas. Se aplica en el análisis y construcción de muchas teorías científicas. Unifica el conocimiento científico en un sistema integral.

❖ **Modelación:** Este método permite la creación de modelos, (propuestas, alternativas, estrategias, etc.) que visualiza una reproducción simplificada de la realidad, que consiente descubrir y estudiar nuevas relaciones y cualidades del objeto de estudio. La modelación es justamente el proceso mediante el cual se crean modelos con vistas a investigar la realidad.

### **Métodos empíricos:**

❖ **Entrevista:** Se utilizará la entrevista como una conversación planificada con los clientes, para obtener información acerca del problema en cuestión. Su uso constituye un medio para el conocimiento cualitativo de las características particulares de un proceso y puede influir en el posterior análisis y diseño del producto de software, que informatizará la supervisión del proceso en sí.

❖ **Método de la observación:** La observación científica es la percepción planificada dirigida a un fin y relativamente prolongada de un hecho o fenómeno. Se realiza de forma consciente y orientada a un objetivo determinado. Dentro del Módulo de Supervisión se hace uso de este método para la síntesis de los flujos de trabajo dentro de las oficinas de CIRP.

El presente documento está compuesto por capítulos, enumerados del 1 al 4, los cuales han sido desarrollados partiendo de los estudios realizados y de la observación de las tareas específicas de los procesos que se despliegan, como parte de la entidad.

A continuación se describe detalladamente el contenido que se abordará en cada uno:

❖ **Capítulo 1: Fundamentación Teórica:** Se aborda una panorámica del estado del arte a enfrentar dentro del problema específico a cumplir. Así como la descripción y fundamentación de las herramientas y tecnologías que se manejarán para darle solución al objetivo trazado. El estudio de este capítulo es opcional si el lector domina los aspectos tratados en este.

❖ **Capítulo 2: Características del sistema:** Se realiza el modelado de negocio analizando sus componentes y cómo interactúan entre sí, poniendo en claro la situación actual a la que se enfrenta el cliente. Muestra los requisitos funcionales y no funcionales, así como el modelo de mejora del proceso, sentando de esta manera las funcionalidades básicas del sistema y la forma de alcanzar una solución óptima.

❖ **Capítulo 3: Análisis y Diseño:** se muestran los flujos de trabajo a seguir y las clases que generará el sistema propuesto incluyéndose en este, los patrones de diseño, servicios que sustentan el software y las clases que tendrán persistencia en la base de datos.

❖ **Capítulo 4: Implementación y Prueba:** Se describe la solución propuesta, donde se analizan los elementos fundamentales que intervienen en la solución de la aplicación, la relación de todos los componentes técnicos que conformarán el sistema, logrando finalmente el despliegue del software luego de haberse probado según los métodos establecidos.



### Capítulo 1: Fundamentación Teórica

#### Introducción

En el presente capítulo se aborda el estudio de algunas de las aplicaciones que supervisan y controlan procesos, en especial, aquellos que sus conceptos se relacionan con los definidos para el Módulo de Supervisión y Control del Sistema Único de Identificación Nacional (SUIN). Además se expondrán las tecnologías de modelado e implementación de las cuales se hizo uso para el desarrollo del proyecto.

#### 1.1 Antecedentes de los procesos de supervisión y control en las Oficinas de CIRP

Los sistemas de identidad, seguros y confiables, garantizan en gran por ciento la seguridad de las naciones y traen consigo un importante impacto en las actividades de la sociedad.

En la actualidad los procesos referentes a la obtención del documento de identidad en Cuba cuentan con escasa informatización, provocando con esto dificultad en la interrelación de informaciones que se encuentran dispersas en sistemas encargados de funciones diferentes cada uno. La obtención de los documentos de identidad no está respaldada por un proceso seguro y la tramitación, emisión y manipulación es débil aun cuando existen medidas para ello.

En lo referente al control y supervisión de la información que de la persona se obtiene para realizar un trámite, no se cuenta con ningún proceso automatizado que permita arreglar en un tiempo adecuado las fallas. Los procesos se realizan de forma manual trayendo consigo errores en los datos, la aceptación de documentos legales es de poca fiabilidad y de forma general existe poca confirmación de la veracidad del trámite en cuestión.

#### 1.2 Conceptos asociados al dominio del problema

##### Trámite

Un trámite es un proceso, diligencia, solicitud o entrega de información que las personas hacen frente a una dependencia, bien sea para cumplir obligaciones o para obtener beneficios o servicios. Es un conjunto o serie de pasos o acciones reguladas por el Estado, que deben efectuar los usuarios para adquirir un derecho o cumplir con una obligación prevista o autorizada por la ley. Los trámites permiten adelantar procesos académicos o administrativos de manera sencilla y eficiente y por lo general los trámites producen documentos. (1)

*Ej.:* Los trámites de obtención de los documentos de identidad, pues una vez obtenido el mismo facilita la identificación de la persona en cualquier ambiente social y lo identifica unívocamente ante cualquier situación.

### **Documento de identificación**

Los documentos de identificación, también llamados documentos nacionales de identidad (DNI) o cédula de identidad, son emitidos por una autoridad administrativa competente para permitir la identificación personal de los ciudadanos residentes en el país, así como los que también no son residentes. Constituyen un documento único de identificación personal e intransferible extendido por una nación a sus ciudadanos para la identificación. Estos usualmente poseen la fotografía del portador, su firma, nombre, nacionalidad, fecha de nacimiento, así como otros elementos que contribuyen a la identificación<sup>1</sup>.

### **Proceso de supervisión y control de trámites**

El proceso de supervisión y control de trámites describe la revisión de datos, documentos e imágenes asociados a un trámite determinado. Se refiere a la comprobación de los datos que se obtienen de la persona que solicita el trámite, al chequeo de la legitimidad de los documentos legales que se deben presentar en el proceso y la calidad de las imágenes de firma digital, huella dactilar y foto de la persona que garantizan una identidad única.

*Ej.* En el caso de sistemas de información podemos encontrar el Sistema SAIME que con la supervisión que lleva a cabo pretende aprobar documentos y con esto imprimir el expediente de la persona y enviar el trámite a su posterior proceso. En el siguiente epígrafe se analizarán soluciones existentes que en la actualidad gestionan procesos de este tipo.

#### **1.3 Análisis de otras soluciones existentes**

En la actualidad muchas son las empresas e instituciones que hacen uso de sistemas que supervisan y controlan procesos en sí. A continuación se mencionan algunos de ellos:

##### **1.3.1 Servicio Administrativo de Identificación Inmigración y Extranjería**

A raíz de los resultados positivos de la *Misión Identidad* en la República Bolivariana de Venezuela, donde se le otorgó la identidad a millones de ciudadanos, surge el proyecto de Transformación de los

---

<sup>1</sup> DNI — Ministerio del Interior de Argentina

procesos de Identificación, Migración y Extranjería, entre cuyos objetivos se encontró la creación de un nuevo sistema para la gestión de la Identificación, Migración y Extranjería (SAIME).

El módulo de Supervisión del SAIME permite conocer si el chequeo dactiloscópico fue efectivo o no. Muestra los documentos y datos que han sido captados al inicio del proceso y brinda la posibilidad de comprobar su validez, teniendo en cuenta los documentos físicos presentados por los ciudadanos. Reúne varias funcionalidades para el control y supervisión de trámites, que serán descritas a continuación:

### **Planilla de Control**

La planilla de control tiene como objetivo que el supervisor verifique los datos captados durante el inicio del proceso y así comprobar si ha ocurrido algún error en los datos básicos de la persona, que son los que se imprimen en dicha planilla. Se imprime una vez que el ciudadano concluye los pasos de captación de datos e imágenes según el trámite que realice.

### **Aprobación de Datos**

Se muestran los datos que han sido recogidos por el sistema durante el trámite en la oficina. Brinda la posibilidad de comprobar la validez y corrección de esta información, teniendo en cuenta los documentos presentados por los ciudadanos. El supervisor que realiza esta tarea es, a su vez, el encargado de autorizar los datos presentados como suficientes para identificar al ciudadano.

### **Aprobación de Documentos**

Se muestran los documentos que han sido captados por el operador durante el proceso. Brinda la posibilidad de comprobar si son válidos, teniendo en cuenta los documentos presentados por los ciudadanos en correspondencia con el trámite. El supervisor que realiza esta tarea es, a su vez, el encargado de autorizar los documentos presentados como suficientes para identificar al ciudadano.

### **Aprobación de Imágenes**

Para la aprobación de imágenes se muestran los ciudadanos pendientes, y se selecciona a aquel al que se le aprobarán. A partir de esta selección, si las imágenes son correctas y cumplen con parámetros preestablecidos, se aprueban en la opción Aprobar imágenes; en caso contrario, se indica la corrección.

### Otras funciones de supervisión

Entre los diferentes subsistemas que trabajan en conjunto para lograr la identificación de las persona, se encuentra un Subsistema de Irregularidades AFIS<sup>2</sup>, que se pone de manifiesto cuando, al chequear la identidad del ciudadano, se recibe una respuesta no esperada. El proceso forma parte de la actividad de apoyo a la Verificación de Identidad que se origina en las oficinas cuando se recibe una respuesta no esperada del AFIS. El trámite de irregularidades se encuentra estructurado por un conjunto de pasos ordenados: Captación de datos, Captación de imágenes, Chequeo dactiloscópico y Supervisión. (2)

#### 1.3.2 EMIPAS: Sistema Cubano de Emisión de Pasaportes

El sistema EMIPAS de emisión de pasaportes es un software que ha sido desarrollado teniendo en cuenta los elementos que se han establecido internacionalmente para la emisión de pasaportes de lectura mecánica. Garantiza:

- ❖ Personalización de Pasaportes de Lectura Mecánica.
- ❖ Código de barra bidimensional para datos biométricos y alfanuméricos.
- ❖ Control de Calidad y Supervisión.
- ❖ Mecanismos de seguridad que personalizan los roles de cada usuario con acceso al sistema.
- ❖ Control de los inventarios de los pasaportes y el estado de cada documento personalizado.
- ❖ Disposición de los medios para la Administración y Mantenimiento, tanto local como a distancia.
- ❖ Conocimiento del funcionamiento del sistema en cada oficina emisora desde la base central.
- ❖ Posibilidad de verificar a cada persona que solicita pasaporte contra la Lista de Impedimento.

Los sistemas de seguridad se extienden hasta el momento de recoger el documento haciendo supervisión en todo momento de los procesos que lleva a cabo para evitar falsificaciones. El solicitante tendrá que verificar su identidad a través de sus huellas digitales en un escáner, entre otros procesos de supervisión. El proceso de registro será con patrón biométrico, fotografía, captura de huellas y firma de forma digital, lo que conlleva e implementa varios servicios de confiabilidad y verificación en el proceso de adquisición del pasaporte. (3)

---

<sup>2</sup> *Automatic Fingerprint Identification System*

### 1.3.3 Sistema de Gestión Académica Akademos

El Sistema Akademos constituye un elemento de gran importancia para el buen funcionamiento de las actividades referentes a la docencia en la Universidad de las Ciencias Informáticas (UCI). Es además una herramienta de trabajo que permite hacer más eficientes los procesos y optimizar sus recursos. Akademos está dividido en seis módulos que abarcan todas las actividades involucradas en la gestión docente. Estos módulos interactúan entre sí para llevar a cabo cada una de las tareas que automatiza el sistema. De estos el que compete al estudio del presente capítulo es el de Matrícula puesto que sus procesos presentan similitudes con las proyecciones de desarrollo del Módulo de Supervisión y Control del SUIN.

#### Procesos que engloba el subsistema matrícula

Cuando a un estudiante se le aprueba la entrada a la Universidad inmediatamente pasa al registro de inscripción de la misma. Luego se dirigen, una vez ingresados, a oficializar la matrícula y en ese momento se hace un completamiento de los datos que necesita la institución sobre el estudiante. La Secretaria Docente se encarga de presentar la información que se ha obtenido de él y se hace un cotejo de los datos para validar que no hay errores y garantizar con ello seguridad en la identidad de los estudiantes y sus parámetros docentes.

Cuando concluye el plazo de matrícula, se crean los grupos docentes y se ubican los estudiantes en los mismos. Existen otros procesos como son los traslados, bajas, licencia de matrícula, reingreso, en el caso de la universidad cubana que se realizan de la misma manera en todos los centros universitarios del país. Todos estos movimientos tienen algo similar, y es que para poder realizarlos se deben llenar una serie de documentos que luego se archivan en el expediente académico del estudiante constituyendo un respaldo legal del proceso que se realiza.

Estos procesos son supervisados por la Secretaria Docente en conjunto con el docente, quien se encarga de validar mediante su firma que los datos registrados en los archivos de la secretaría docente son los correctos. (4)

#### 1.3.3.1 Inconvenientes de los sistemas analizados

Todas las soluciones antes mencionadas permiten la supervisión de procesos de forma general. Cada evento que realizan tiene pautas diferentes y se rige por procedimientos ya definidos siguiendo las normativas del ambiente en que se utiliza. A continuación se muestran las inconformidades

detectadas y por las cuales se ha decidido la creación de un nuevo Módulo de Supervisión y Control para el SUIN que contenga las funcionalidades requeridas por el cliente.

❖ **Sistema SAIME:** Este sistema fue pensado y modelado según las especificaciones dictadas por el cliente y las particularidades sociales y legales de Venezuela, país en el cual se desplegó el mismo. No fue elaborado para uso de manera general en cualquier ambiente por las particularidades del negocio.

❖ **Sistema EMIPAS:** El sistema EMIPAS presentaba como principal ventaja el haber sido desarrollado por especialistas cubanos, sin embargo, no cumple con los requisitos del Módulo de Supervisión y Control pues enfoca la supervisión al proceso de personalización del pasaporte. El subsistema del SUIN debe hacer una supervisión completa de datos, documentos e imágenes relativas a cualquier trámite. Partiendo de esto, se puede inferir que el sistema EMIPAS no condiciona su uso para las actividades que se requieren.

❖ **Sistema Akademos:** El Sistema de Gestión Académica Akademos, como su nombre lo indica, se refiere al control de procesos docentes y no a la identificación de personas como los otros sistemas tratados anteriormente. Esto sin dudas lo hace fácil de descartar pues sus funcionalidades no aportan destino ninguno al SUIN. Sin embargo, la supervisión que lleva a cabo en las actividades de matrícula y rematrícula, sirven de guía para entender los procesos y las facilidades que provee el control sobre los eventos.

### 1.4 Ambiente de desarrollo

El proceso de implementación de una solución de software implica metodologías, herramientas de modelado de procesos, de desarrollo, tanto para la aplicación como para la base de datos, y otras más que unidas conforman el sistema con sus capacidades. La solución del Módulo de Supervisión y Control marcha sobre la base del proyecto Identificación, Migración y Extranjería de la República de Cuba, el cual ha realizado estudios sobre las herramientas y metodologías de desarrollo a utilizar apoyándose siempre en las facilidades que ofrecían para el ambiente de desarrollo. A continuación se hace un desglose detallado de cada una de ellas.

### 1.4.1 Desarrollo basado en procesos

El desarrollo tecnológico, propiciado especialmente por las TIC<sup>3</sup>, ha provocado que las empresas de desarrollo de software busquen eficacia y eficiencia para continuar compitiendo en la nueva sociedad de la información.

En los últimos años los desarrolladores de software han enfocado a procesos la implementación de sistemas informáticos. A diferencia de lo que puede ser un proceso rígido con pocas posibilidades de variar, los procesos de *workflow*<sup>4</sup> son propicios al cambio. Esta característica se debe principalmente a la alta participación de personas en tales procesos las cuales pueden ser reubicadas, ascendidas o suprimidas en el organigrama de una organización implicando de esta manera la variación temporal o definitiva de un determinado proceso. También la búsqueda constante de mejora en los tiempos de procesos por parte de la organización hace que los mismos sean modificados para eliminar “cuellos de botella” detectados luego de modelarlos. La programación orientada a procesos facilita el cambio en cualquiera de las actividades del negocio que en un futuro varíe y necesite reestructurarse en el sistema informático. (5)

### 1.4.2 Metodología de desarrollo

*MSF*<sup>5</sup> for *CMMI*<sup>6</sup> se acerca a una nueva dimensión donde el desarrollo integrado fortalece el desarrollo de aplicaciones. Con el uso de la herramienta Visual Studio 2008 se han unido dos metodologías que organizan y controlan los procesos de desarrollo de software permitiendo la supervisión de los códigos trabajados así como la documentación generada como parte del proceso de desarrollo de software. Esta metodología establece un proceso disciplinado de desarrollo de software que lo hace más predecible y eficiente. Los miembros del equipo pueden asumir uno o varios roles en función de las necesidades del ambiente de desarrollo ejecutando actividades que producen resultados de valor observable como son la documentación, código fuente, plan de proyecto y otros que constituyen la base del proyecto.

Cada miembro del equipo de trabajo juega un papel fundamental en los ciclos de vida de MSF for CMMI incluyéndose visión, producción, uso y mantenimiento. En el trabajo conjunto con el *Team*

---

<sup>3</sup> *Tecnologías de Informática y Comunicaciones*

<sup>4</sup> *De la definición en inglés de Flujo de Trabajo*

<sup>5</sup> *Microsoft Solution Framework*

<sup>6</sup> *Capability Maturity Model Integration*

*Model*<sup>7</sup> se orientan las actividades, se chequean y se hace un balance de la calidad con el propósito de cumplir con las expectativas del cliente.

La integración de MSF con la herramienta *Team System* ofrece soporte para un desarrollo rápido e interactivo que además permite aprendizaje continuo y refinamiento. Los ciclos e iteraciones hacen un enfoque a la reducción de errores permitiendo así minimizar riesgos en la producción. MSF for CMMI *Process Improvement*<sup>8</sup> define cinco fases para el ciclo de vida del proyecto, que encapsulan actividades y *workstream*<sup>9</sup>. Cada fase comienza cuando aún la anterior no ha llegado a su fin permitiendo así la continuidad o no del trabajo en función del avance del proceso de desarrollo. (6)

### 1.4.3 Notación para el Modelado de Procesos de Negocio

#### Modelado de Procesos de Negocio

Una vez definida la metodología de desarrollo de software el equipo de trabajo debe comenzar a modelar los procesos de negocio del área donde el sistema será desplegado. Las herramientas que facilitan esta actividad visualizan, especifican, construyen y documentan el sistema informático en progreso permitiendo la captura de información de larga vida que forma parte de los artefactos generados por el proyecto en sí.

*BPMN*<sup>10</sup> es un nuevo estándar de modelado de los asuntos de negocio, donde se presentan gráficamente las diferentes etapas del proceso del mismo. La notación ha sido diseñada específicamente para coordinar la secuencia de eventos y los mensajes que fluyen entre los diferentes procesos participantes.

BPMN cubre casi totalmente los patrones de workflow con lo cual se le supone una gran expresividad a la hora de especificar procesos. (7)

#### UModel – Herramienta UML para el modelado y desarrollo de aplicaciones.

Altova UModel 2009 se considera como el punto de salida para el desarrollo de software de éxito. Diseña visualmente modelos de aplicaciones en BPMN y *UML*<sup>11</sup>, genera código Java, C#, o VB .NET y documentación del proyecto y realiza ingeniería inversa de los programas existentes pasándolos a

---

<sup>7</sup> Del inglés *Modelo de Equipo*

<sup>8</sup> Del inglés *Proceso Mejorado*

<sup>9</sup> Del inglés *Secuencia de Actividades de Trabajo*

<sup>10</sup> Siglas del inglés *Business Process Management Notation (Notación para el Modelado de procesos de Negocio)*

<sup>11</sup> Siglas del inglés *Unified Model Language*



diagramas UML 2. Esta herramienta hace el diseño visual de software práctico para cualquier proyecto permitiendo, de manera simple y asequible, dibujar en UML. Combina una rica interfaz visual con funciones de usabilidad superiores para ayudar a nivelar la curva de aprendizaje de UML. Las características de UModel 2009 para el desarrollo de software basado en las capacidades de modelado avanzado son:

- ❖ Soporte para los 14 tipos de diagramas UML
- ❖ Modelado de esquemas XML<sup>12</sup> en diagramas UML
- ❖ Diagramas de proceso de negocio (BPMN)
- ❖ Generación de código fuente en lenguajes Java, C#, y VB.NET
- ❖ Ingeniería inversa de código fuente y ficheros binarios Java, C# y VB.NET
- ❖ Sincronizado de modelo y código a través de ingeniería de ida y vuelta
- ❖ Crea diagramas de secuencia desde el código fuente de la ingeniería inversa
- ❖ Generación de documentación personalizable de proyecto
- ❖ Compartir subproyectos para colaboración o reutilización
- ❖ Capas de diagramas con visibilidad selectiva
- ❖ Hipervínculos entre diagramas, documentos, o páginas Web
- ❖ Integración con sistemas de control de versiones
- ❖ Estrecha integración con Visual Studio y Eclipse (8)

#### 1.4.4 Herramienta para la gestión de Base de Datos

Los Sistemas Gestores de Bases de Datos (SGBD) proveen facilidades para la manipulación de grandes volúmenes de información, de aquí la importancia que tienen para los sistemas informáticos pues manejan, de forma clara, sencilla y ordenada datos que posteriormente se convertirán en información de relevancia. (9)

---

<sup>12</sup> Siglas del inglés Extensible Markup Language

### Oracle 11gR213 como SGBD

Oracle Database 11g proporciona funcionalidades que garantizan alto rendimiento, alta escalabilidad, fiabilidad y seguridad mediante el uso de plataformas *grid*<sup>14</sup>, asegurando a la vez altos niveles de calidad de servicio e incrementos de la flexibilidad de negocio, reduciendo además los costes de explotación. Se pueden resolver las problemáticas de negocio más exigentes en todas las áreas, incluyendo aplicaciones transaccionales, de inteligencia de negocio y de gestión de contenidos.

Oracle Database 11g incluye funcionalidades que permiten hacer pruebas de cambios en aplicaciones simulando las cargas reales generadas por los usuarios en los entornos de producción. *Real Application Testing*<sup>15</sup> permite reducir de manera drástica los tiempos, riesgos y costes derivados de la implantación de cambios, asegurando que las aplicaciones se comportarán de manera adecuada y predecible tras las modificaciones. Con Real Application Testing los clientes ganan en flexibilidad puesto que pueden responder de manera más efectiva a los requerimientos cambiantes del negocio y hacer una gestión del cambio más efectiva. (10)

- ❖ Archivos seguros: gestión eficiente y segura de todo tipo de datos.
- ❖ Arquitectura de máxima disponibilidad.
- ❖ Gestión de datos XML.
- ❖ Mejoras en los pool de conexiones y en los caches de resultados.
- ❖ Mejoras en el desarrollo de aplicaciones.
- ❖ Mejoras en la automatización y en la auto-administración de la base de datos.

#### 1.4.5 Herramientas para el entorno de desarrollo integrado

La meta fundamental de los desarrolladores de software en la actualidad es crear aplicaciones cada vez mejores y en el menor tiempo posible (11). Se persigue como objetivo fundamental adquirir herramientas que permitan centralizar actividades con el fin de tener mayor comunicación, colaboración, procesos coherentes y al final, alcanzar el término exitoso de la tarea.

*Microsoft Visual Studio Team System 2008* es una nueva herramienta para los desarrolladores que facilita lo antes mencionado, pues presenta nuevas funcionalidades como el *IntelliSense* para

---

<sup>13</sup> Oracle Versión 11g Release 2

<sup>14</sup> Infraestructura que permite la integración y el uso colectivo de ordenadores de alto rendimiento, redes y bases de datos que son propiedad y están administrados por diferentes instituciones

<sup>15</sup> De la definición en inglés de Aplicaciones de Pruebas Reales

*JavaScript*, que ayudará a los programadores de Aplicaciones Web a realizar sus proyectos mucho más rápido; también el hecho de tener la posibilidad de cambiar entre versiones del Framework .NET ayuda a las empresas a elegir el tipo de estructura que ocuparán sus nuevos sistemas. (12)

### **Visual Studio Team System 2008**

*Visual Studio Team System* facilita la comunicación y la colaboración del equipo de desarrollo, proporcionando un repositorio unificado de todos los datos del proyecto, junto con las herramientas para definir, ejecutar y automatizar los procesos deseados. En su conjunto de herramientas integra *Team Foundation Server 2008*, que permite dar soporte al código fuente y al control de versiones, así como dar seguimiento a los elementos de trabajo, construir y automatizar los controles de calidad. Los artefactos, datos de proyectos, control de código fuente, creación y prueba se guardan en un almacén de datos, mientras que los paneles e informes eficaces proporcionan datos de las tendencias de historial, capacidad de seguimiento total y visibilidad en tiempo real para la calidad y el progreso del objetivo de la empresa. (12)

Ofrece una amplia gama de herramientas para todas las fases del desarrollo de software, incluidos la creación, la prueba, la implementación, la integración y la administración. Además, permite a los desarrolladores comunicarse mediante máquinas, servidores, la Web y dispositivos móviles.

### **Framework .NET 3.5**

Este *framework*, está diseñado para aprovechar las ventajas que ofrece el nuevo sistema operativo "Windows Vista" a través de sus subsistemas "*Windows Communication Foundation*" (WCF) y "*Windows Presentation Foundation*" (WPF).

A las mejoras de desempeño, escalabilidad y seguridad con respecto a la versión anterior, se agregan entre otras, las siguientes novedades:

- ❖ La mejora en las capacidades de Pruebas Unitarias permiten ejecutarlas más rápido independientemente de si lo hacen en el entorno *IDE*<sup>16</sup> o desde la línea de comandos. Se incluye además un nuevo soporte para diagnosticar y optimizar el sistema a través de las herramientas de pruebas de Visual Studio. Con ellas se podrán ejecutar perfiles durante las pruebas para que ejecuten cargas, prueben procedimientos contra un sistema y registren su comportamiento; y utilizar herramientas integradas para depurar y optimizar.

---

<sup>16</sup>Siglas del inglés *Integrated Development Environment* (Ambiente de Desarrollo Integrado)

- ❖ NET 3.5 incluye biblioteca ASP.NET AJAX para desarrollar aplicaciones web más eficientes, interactivas y altamente personalizadas que funcionen para los navegadores más populares y utilicen las últimas tecnologías y herramientas Web, incluyendo Silverlight y Popfly. (13)

### ASP. NET

ASP.NET es la parte del .NET Framework dedicada al desarrollo web. A través del servidor web (IIS)<sup>17</sup> las aplicaciones ASP.NET se ejecutarán bajo el CLR<sup>18</sup> y se podrán usar el conjunto de clases del .NET Framework para desarrollarlas, obteniendo así una versatilidad y potencia que versiones anteriores no tenían.

También son destacables los servicios web, que permitirán la comunicación a través de internet entre diferentes ordenadores, incluso entre distintos sistemas. Permite crear aplicaciones en el más clásico de los sentidos. (14)

ASP.NET se ha construido bajo los siguientes principios:

- ❖ Facilidad de desarrollo
- ❖ Alto rendimiento y escalabilidad
- ❖ Mejorada fiabilidad
- ❖ Fácil distribución e instalación

### Csharp (C#) como lenguaje de programación

C# es un lenguaje de programación que está incluido en la Plataforma .NET y corre en CLR. Aunque para la plataforma .NET es prácticamente posible programar en cualquier lenguaje, el C# es el lenguaje de propósito general diseñado por Microsoft para ser utilizado en ella, por lo que programar usando C# es mucho más sencillo e intuitivo que hacerlo con cualquiera de los otros.

Entre sus principales características se destacan:

- ❖ C# provee el beneficio de un ambiente elegante y unificado.
- ❖ No soporta herencia múltiple, solamente el *runtime*<sup>19</sup>.NET permite la herencia múltiple en la forma de interfaces, las cuales no pueden contener implementación.

---

<sup>17</sup> Siglas del inglés *Internet Information Service*

<sup>18</sup> Siglas del inglés *Common Language Runtime*

<sup>19</sup> De la definición en inglés de tiempo real

- ❖ El manejo de errores está basado en excepciones.
- ❖ Soporta los conceptos como encapsulación, herencia y polimorfismo de la programación orientada a objetos.
- ❖ No existen funciones globales, variables o constantes. Todo debe ser encapsulado dentro de la clase, como un miembro de la instancia (accesible vía una instancia de clase) o un miembro estático (vía el tipo).
- ❖ Los métodos que se definen en las clases son por defecto no virtuales (no pueden ser sobrescritos al derivar clases).
- ❖ Solamente se permite una clase base, si se requiere herencia múltiple es posible implementar interfaces.
- ❖ No es posible utilizar variables no inicializadas.
- ❖ No es posible hacer el casteo de un entero a un tipo de referencia (objeto). (15)

### Windows Workflow Foundation

Un workflow es un conjunto de actividades guardadas como un modelo que describe un proceso del mundo real. El trabajo pasa a través del modelo desde el principio hasta el final, y las actividades pueden ser ejecutadas por personas o por funciones del sistema. Workflow provee una forma de describir el orden de ejecución y la dependencia de las relaciones entre las piezas de corta o larga duración. (16)

Es una plataforma que permite a los usuarios crear un flujo de trabajo en sus aplicaciones. Consiste en un espacio de nombres, un motor de flujos de trabajo en proceso, y diseñado para Visual Studio, y puede ser utilizado en escenarios simples, como sería mostrar controles UI<sup>20</sup> basados en entradas de usuarios, o escenarios complejos como con los que se tropieza en las grandes empresas, como son el procesamiento de pedidos y control de inventario. (17)

### Capa de Acceso a Datos

En la actualidad los desarrolladores de software buscan las herramientas que le permitan crear aplicaciones de acceso a datos programadas en un modelo de la aplicación conceptual en lugar de hacerlo directamente con un esquema de almacenamiento relacional. El objetivo fundamental que se persigue es el de reducir la cantidad de código y mantenimiento que se necesita para las aplicaciones orientadas a datos. Esto ofrece grandes ventajas que favorecen al desarrollo como por

---

<sup>20</sup> Siglas del inglés *User Interface*

ejemplo el hecho de que las aplicaciones están libres de dependencias de codificación rígida de un motor de datos o de un esquema de almacenamiento, también las asignaciones entre el modelo conceptual y el esquema específico de almacenamiento pueden cambiar sin tener que cambiar el código de la aplicación.

### **Entity Framework**

Entity Framework es un conjunto de tecnologías de ADO.NET que facilitan el desarrollo de aplicaciones de software orientadas a datos. Permite a los programadores trabajar con datos en forma de objetos y propiedades específicos del dominio, por ejemplo, con clientes y direcciones, sin tener que pensar en las tablas de las bases de datos subyacentes y en las columnas en las que se almacenan estos datos. Para ello, se eleva el nivel de abstracción en la que los programadores pueden trabajar al tratar con datos y se reduce el código requerido para crear y mantener las aplicaciones orientadas a datos. Dado que Entity Framework es un componente de .NET Framework, las aplicaciones de Entity Framework se pueden ejecutar en cualquier equipo en que esté instalado .NET Framework 3.5 Service Pack 1 (SP1). A través de la infraestructura de Servicios de objeto de Entity Framework, ADO.NET expone una vista conceptual común de los datos, incluidos los datos relacionales, como objetos del entorno .NET. Esto hace que la capa de objetos sea un objetivo ideal para la compatibilidad con LINQ<sup>21</sup>. (18)

### **LINQ como lenguaje de enlace entre los datos y el lenguaje de programación**

Language-INtegrated Query (LINQ) es un conjunto de características en Visual Studio 2008 que agrega eficaces capacidades de consulta a la sintaxis de los lenguajes C# y Visual Basic. LINQ incluye patrones estándares y de fácil aprendizaje para consultar y actualizar datos, y su tecnología se puede extender para utilizar potencialmente cualquier tipo de almacén de datos. Visual Studio 2008 incluye ensamblados de proveedores para LINQ que habilitan su uso con colecciones de .NET Framework, bases de datos de SQL Server, conjuntos de datos de ADO.NET y documentos XML.

Es un modelo de programación que simplifica y unifica la implementación de acceso a cualquier tipo de dato. LINQ no impone a usar una arquitectura específica más bien facilita la implementación de varias arquitecturas existentes para acceso a datos.

---

<sup>21</sup> Siglas del inglés Language-INtegrated Query

LINQ para SQL<sup>22</sup> es una implementación de O/RM<sup>23</sup> que viene con la nueva versión del .NET Framework y permite modelar bases de datos relacionales con clases de .NET. Se puede consultar bases de datos así como actualizar, añadir y borrar datos de ellas. (19)

Visual Studio 2008 incluye un diseñador de LINQ to SQL para modelado de BD de ahí el porqué de su uso para el desarrollo del SUIN, pues se acopla al entorno de desarrollo integrado que los líderes del proyecto persiguen.

### 1.4.6 Herramienta para el modelado de datos

#### Embarcadero ER/Studio 8.0

Embarcadero ER/Studio se considera el líder en modelado de datos, pues permite documentar y reutilizar los activos en datos. Facilita la ingeniería inversa y optimizar las bases de datos existentes, además, con su uso se gana en productividad y fuerza el cumplimiento de los estándares de la organización. Permite a los usuarios visualizar, analizar y documentar cómo fluyen los datos a través de la organización. Entre las ventajas de su uso se pueden encontrar:

- ❖ Entorno de diseño dirigido por el modelo.
- ❖ Soporte al ciclo de vida completo de las bases de datos.
- ❖ Ingeniería inversa y directa y generación automatizada de código de bases de datos.
- ❖ Gestión de modelos empresariales.
- ❖ Capacidades de comunicaciones empresariales.
- ❖ Almacén de datos y soporte a la integración.
- ❖ Diseños de calidad de bases de datos.
- ❖ Validación de diseño, forjamiento de la integridad referencial y capacidad de planificación y modelado de seguridad. (20)

### 1.4.7 Herramienta para la obtención de reportes

La mayoría de los sistemas de información desarrollados actualmente constan de módulos de reportes que permiten a los usuarios obtener información basada en parámetros específicos. En muchos casos se hace uso de estos para emitir documentos con parámetros específicos que han sido determinados por el usuario. El SUIN, como producto integrador de funcionalidades, incluye

---

<sup>22</sup> Siglas del inglés *Structured Query Language*

<sup>23</sup> Siglas del inglés *Object Relational Mapping (Mapeador de Objetos Relacionales)*

entre sus módulos el de reportes lo cual facilitará el control sobre procesos determinados o usuarios que interactúan con el sistema y a los cuales se les hará una medición de la actividad que realizan.

### **Crystal Reports básico para Visual Studio 2008**

Crystal Reports es una solución de generación de informes que ayuda a los clientes a crear informes flexibles y fiables, e integrarlos en aplicaciones tanto de cliente ligero como completo. Ofrece avanzadas funcionalidades que contribuyen a reducir la proliferación de informes y el trabajo asociado a su mantenimiento, ofrecen una mayor flexibilidad de visualización y ahorran tiempo, gracias a la extraordinaria productividad de sus funciones de diseño. (21)

La solución Crystal Reports consta de:

- ❖ Un potente diseñador de informes
- ❖ Flexibilidad en el desarrollo de aplicaciones
- ❖ Administración y distribución de informes

Crystal Reports destaca funcionalidades cómo:

- ❖ Repositorio: Almacenar objetos de reporte clave en una librería centralizada, para ser reutilizados en otros reportes.
- ❖ Report Application Server<sup>24</sup>: Servicio de procesamiento de reportes basado en Web, para realizar integración de contenido dinámico en aplicaciones Web empresariales.
- ❖ SDKs<sup>25</sup> para .NET, Java<sup>26</sup> y COM: Crear aplicaciones utilizando cualquier plataforma de desarrollo y controlando la visualización, interacción y modificación de los reportes en tiempo de ejecución.
- ❖ Licencias de publicación Web flexibles: Capacidades de procesamiento en cola, que hacen aún mejor el aprovechamiento de sus licencias. (22)

---

<sup>24</sup> Servidor de reportes de aplicaciones

<sup>25</sup> Siglas del inglés Software Development Kit

<sup>26</sup> Lenguaje de programación



### **Conclusiones**

- ❖ La elaboración del capítulo ofrece un extenso y abarcador estudio de las aplicaciones que gestionan procesos en el mundo actual, específicamente, las que supervisan y controlan los mencionados procesos.
- ❖ Se han analizado las soluciones existentes que podrían brindar el servicio que se busca y después de hacer valoraciones juiciosas se ha determinado la necesidad de crear un nuevo software que abarque las especificaciones requeridas por el cliente.
- ❖ Se identifican y fundamentan las tecnologías utilizadas para el desarrollo de la aplicación sentando las bases en las ventajas que ofrecen al desarrollo integrado y de acuerdo a peticiones específicas del cliente.

### Capítulo 2: Características del sistema

#### Introducción

En el presente capítulo se describe cómo se manifiesta el trabajo en las oficinas del Carné de Identidad y Registro de la Población (CIRP). Se detallan los procesos de negocio actuales, y con esto, se identifican los trabajadores que intervienen así como también se especifican las características del sistema propuesto a implementar.

#### Modelo de Negocio

##### 2.1 Descripción del proceso de Supervisión actual

En la actualidad los procesos relacionados con la supervisión de los trámites en las oficinas de CIRP se realizan como parte de la confección del documento de identidad. El primer funcionario, quien es el encargado de la confección del documento, valida los datos evitando así errores en este que afecten la obtención de la identidad por parte de la persona. Al finalizar la confección si no se presentaron errores el funcionario plasma la firma de autorizo del trámite. En la entrega del documento identificativo el funcionario supervisa otra vez la legibilidad y fiabilidad de los datos y documentos emitidos en el proceso garantizando así seguridad en el mismo.

Por la importancia de este flujo se ha decidido separar en un módulo independiente de los demás y se han definido procesos mejorados que harán la tramitación segura y confiable.

##### 2.2 Diagrama del proceso de Supervisión

La actividad de supervisión, como se ha mencionado anteriormente, en este documento se incluye dentro de grandes procesos. Para mejor entendimiento ver **Fig. 1 Diagrama del proceso Confección** y **Fig. 2 Diagrama del proceso Entrega**, donde se especifica en qué parte interviene la supervisión.

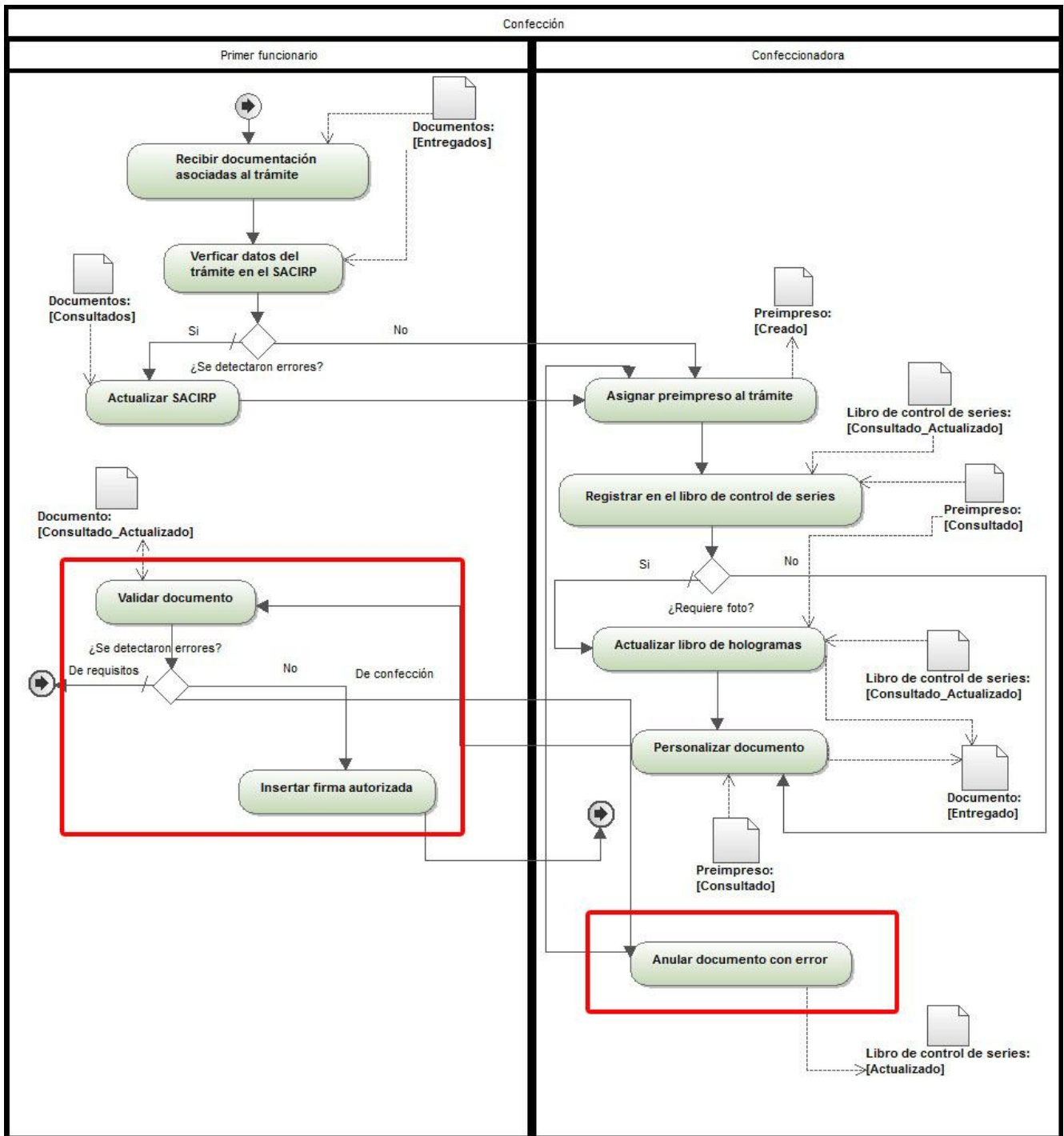


Fig. 1 Diagrama del proceso Confección

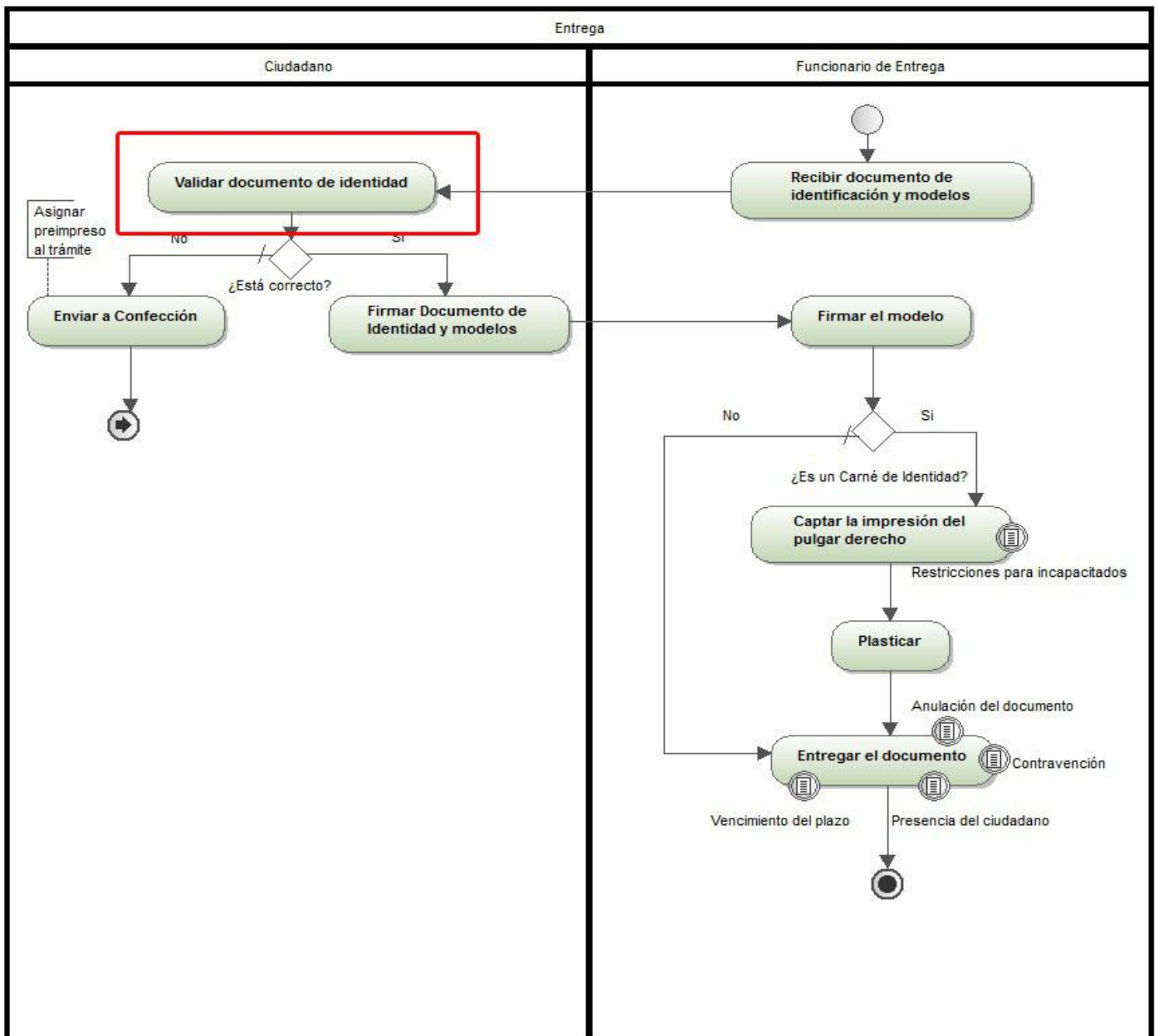


Fig. 2 Diagrama del proceso Entrega

### Descripción de las actividades del proceso de Supervisión

A continuación se describe la actividad Validar documentos la cual se lleva a cabo en el área de Confección. Las demás actividades que incluyen la supervisión se encuentran, en el orden correspondiente, en el **Anexo 1: Descripción de las actividades del negocio**.

### Confección

#### A.1 Validar documento

**Precedencia:** Personalizar documento.

**Rol:** Primer funcionario.

**Entradas:** Documentos.

**Salidas:** -

**Descripción:** El primer funcionario verifica que la documentación esté correcta y en correspondencia con el trámite realizado.

**Bifurcación:** ¿Se detectaron errores?

Si son de confección: **A3:** *Anular documento con error.*

Si son de requisitos: Volver el proceso a Tramitación.

Si no se detectaron errores: **A2:** *Insertar firma autorizada.*

#### 2.3 Especificación de requisitos funcionales

Para el desarrollo de una solución de software se deben detallar las funcionalidades que este debe cumplir según las especificaciones del cliente, buscando como resultado calidad y satisfacción de los mismos. Estas funcionalidades reciben el nombre de requisitos funcionales y según la definición de la *IEEE*<sup>27</sup> *Standard Glossary of Software Engineering Terminology* un requisito funcional es la condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo. Condición o capacidad que tiene que ser alcanzada o poseída por un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente. (23)

##### 2.3.1 Concepción de los modelos de mejora de procesos

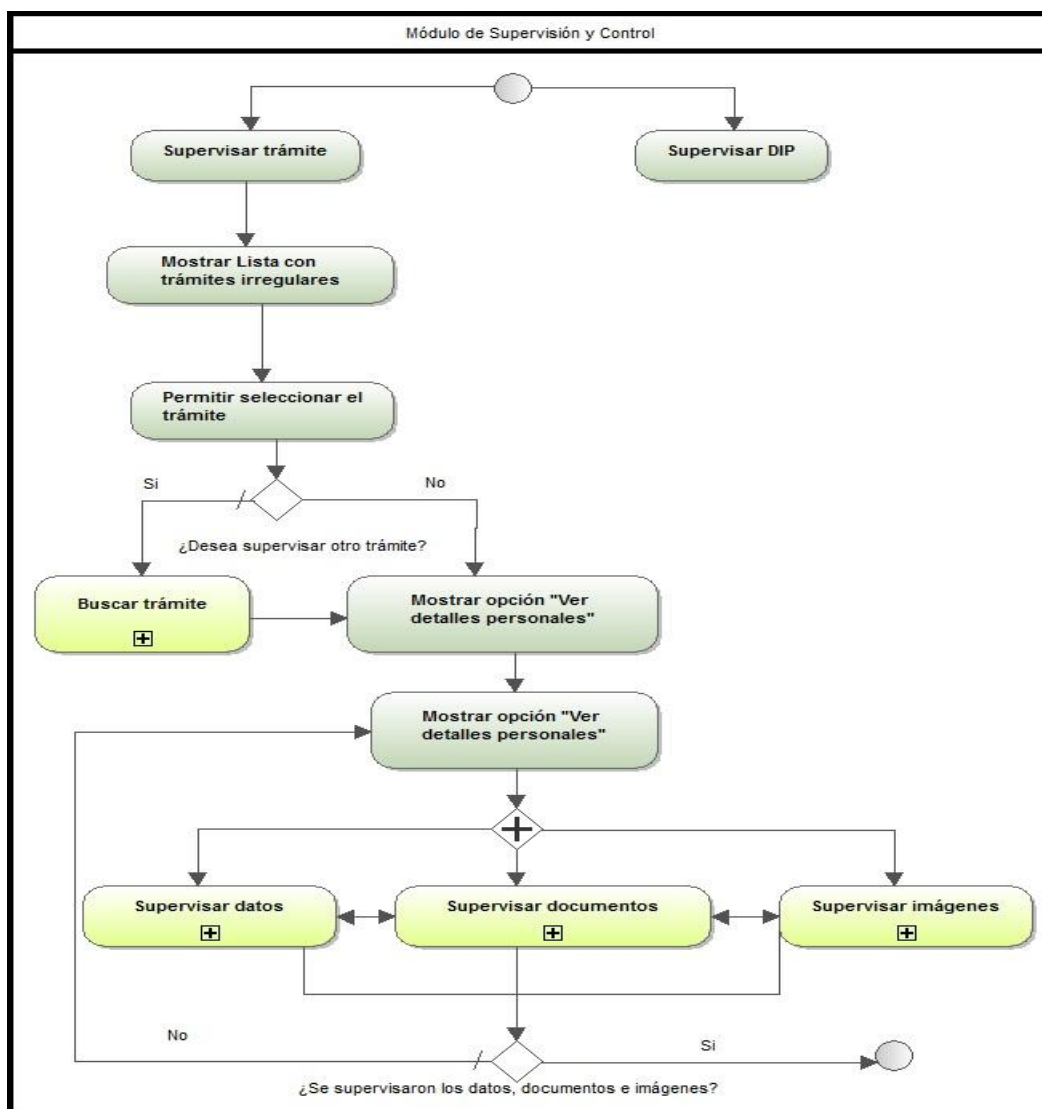
Los autores de este trabajo definen como un proceso en la aplicación, al conjunto de eventos que inician las actividades por parte de los funcionarios que intervienen con el sistema informático. El Módulo de Supervisión y Control, como parte del SUIN, permitirá optimizar los procesos de supervisión y control que van asociados a los distintos trámites que se realizan en las oficinas de CIRP. Esta solución de software brindará las funcionalidades de supervisión a los datos, documentos e imágenes que pertenecen al trámite, búsqueda de trámites, impresión del Documento de Identidad Provisional y visualización del expediente de la persona.

---

<sup>27</sup> *Instituto de Ingenieros Eléctricos y Electrónicos.*

Un modelo de mejora de procesos es un conjunto estructurado de elementos cuyo objetivo es el desarrollo de productos de calidad de manera consistente y predecible. Un modelo indica "Qué hacer", no "Cómo hacer", ni "Quién lo hace". (24)

El modelo de mejora de procesos se lleva a cabo una vez analizado el negocio y como base para el posterior desarrollo del sistema, permitiendo así hacer una muestra del comportamiento del software y facilitando la recuperación de errores que pueda haber en los procedimientos descritos. A continuación se muestran los modelos de procesos correspondientes a cada funcionalidad de la solución:

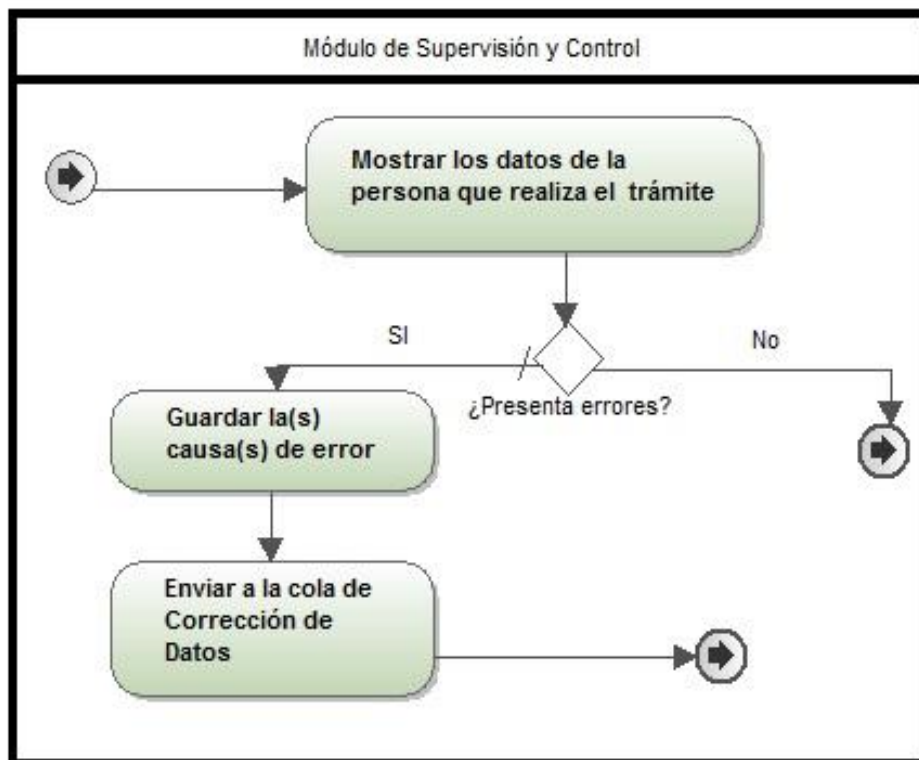


**Fig. 3** Macro Proceso Supervisar Trámite

La supervisión se puede iniciar cuando el funcionario de la oficina de CIRP necesita verificar datos de un trámite que presenta errores y requiere la certificación de su validez. La otra variante para dar inicio al flujo es cuando se realiza la impresión y entrega del Documento de Identidad Provisional (DIP) y este presenta errores en los datos de la persona a la cual corresponde.

### Subproceso Supervisar datos

La supervisión de datos se incluye dentro del proceso Supervisar trámite y se hace con el propósito de corregir errores en los datos obtenidos de la persona a la cual se le está realizando el trámite. Una vez identificados estos errores, el trámite se envía corrección de datos para el arreglo de los elementos que pueden ser modificados. Ver **Fig. 4 Subproceso Supervisar datos**.



*Fig. 4 Subproceso Supervisar datos*

Para mejor comprensión de los subprocesos supervisar documentos e imágenes ver **Anexo 2: Proceso mejorado de Supervisar documentos e imágenes**.

### Proceso Imprimir DIP

La impresión del DIP es otra de las actividades que dan inicio al flujo de supervisión. La persona recibe el Documento de Identidad Provisional una vez terminada la Captura de Datos e Imágenes. Los datos impresos en el documento son verificados por la persona en trámite y de encontrarse algún error se da inicio al flujo de Supervisión permitiendo la reimpresión del DIP una vez corregidos. De no presentar ninguna irregularidad la persona adquiere el DIP y el trámite pasa a Personalización para la confección del documento de identidad. Ver **Fig. 5** Proceso Imprimir DIP.

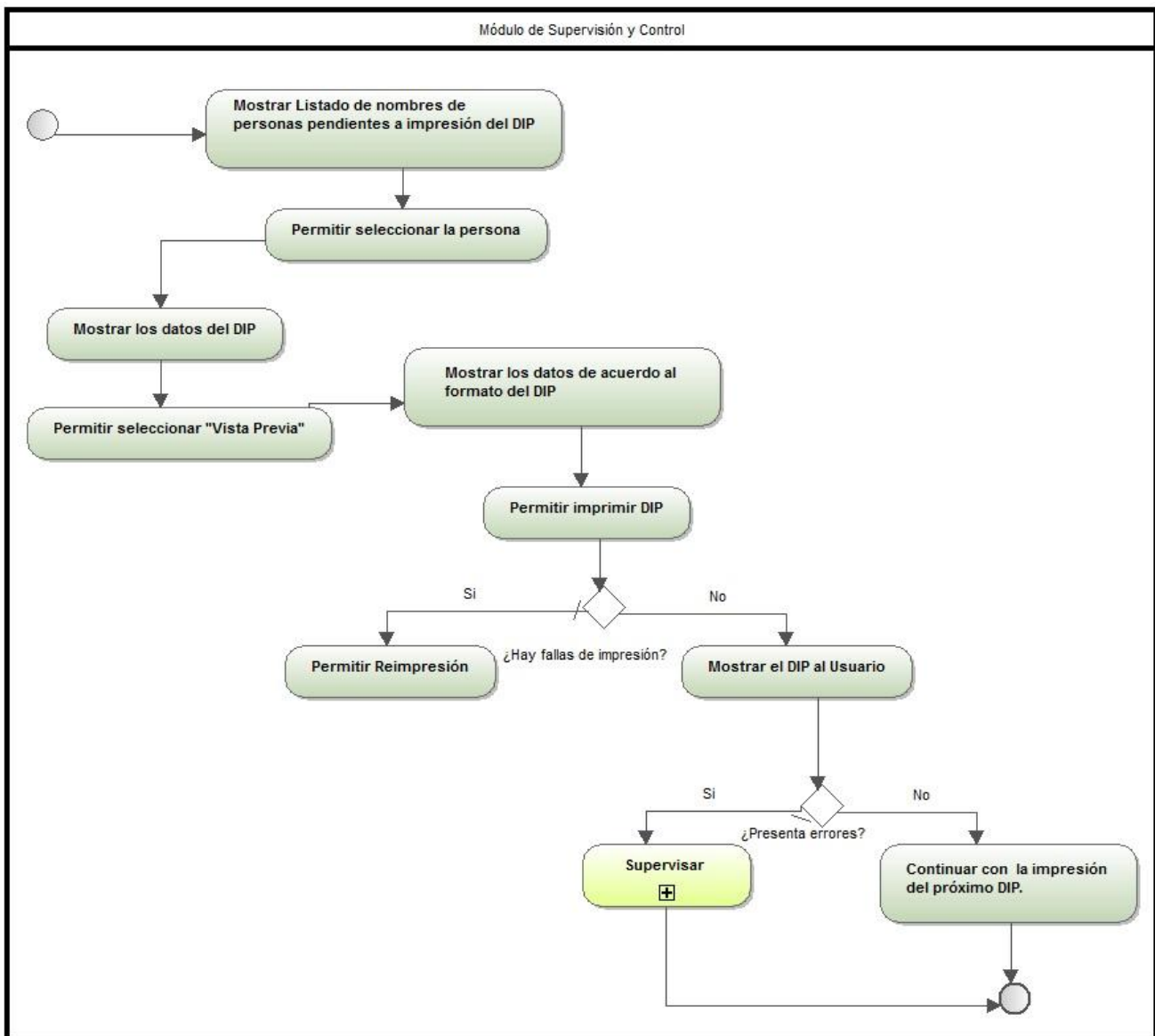
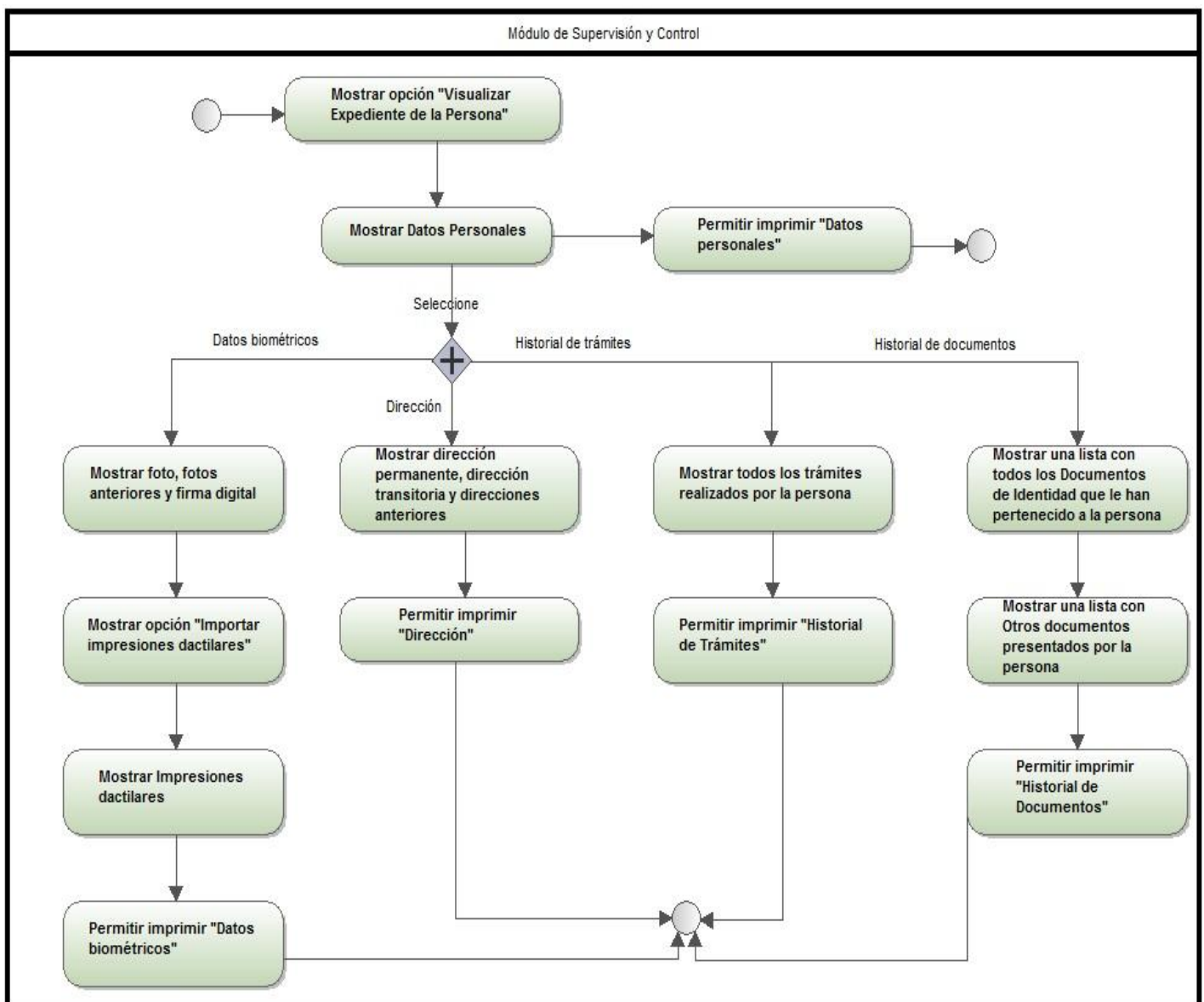


Fig. 5 Proceso Imprimir DIP



**Proceso Visualizar expediente de la persona**

La opción para visualizar el expediente de la persona en el módulo de Supervisión facilita la revisión de todos los procesos que esa persona ha efectuado antes en las unidades de CIRP. Entre los aspectos que muestra se incluyen sus datos personales, trámites realizados con anterioridad, direcciones anteriores y si tiene registros en la Base de Poseedores. Ofrece además funcionalidades para la impresión de cualquiera de los datos que muestra. Ver **Fig. 6** Proceso Visualizar expediente de la persona.



**Fig. 6** Proceso Visualizar expediente de la persona

### 2.3.2 Roles del sistema

Los roles del sistema no son más que aquellos trabajadores del negocio que interactúan con el software y que tienen permisos en el área donde intervienen.

Rol	Objetivo
<b>Funcionario de Supervisión</b>	Es el responsable de la supervisión de los datos, documentos e imágenes asociados a un trámite así como de la búsqueda del mismo. Interactúa con el expediente de la persona y facilita su visualización además de dirigir el control de funcionarios y de trámites.

*Tabla 1 Roles del sistema*

### 2.3.3 Catálogo de requisitos funcionales

#### Módulo de Supervisión y Control

**RF 1.** Permitir la supervisión de datos.

- 1.1 Mostrar todos los datos de la persona.
- 1.2 Permitir seleccionar el tipo de error.
- 1.3 Enviar el trámite a Captación de datos.

**RF 2.** Permitir la supervisión de documentos.

- 2.1 Mostrar listado de documentos captados.
- 2.2 Permitir seleccionar el tipo de error.
- 2.3 Guardar la(s) causa(s) de error.
- 2.4 Si las causas de error son del documento, cancelar el trámite.
- 2.5 Si las causas de error son de datos, enviar el trámite a captura de datos.

**RF 3.** Permitir la verificación de las imágenes captadas (foto, impresiones dactilares, firma)

- 3.1 Mostrar las imágenes asociadas con el trámite que se supervisa.
- 3.2 Mostrar foto y álbum de fotos anteriores.
- 3.3 Si presenta deficiencias permitir guardar la(s) causa(s) de error.
- 3.4 Mostrar imagen Impresiones dactilares.
- 3.5 Si presenta deficiencias permitir guardar la(s) causa(s) de error.
- 3.6 Mostrar imagen firma.
- 3.7 Si presenta deficiencias permitir guardar la(s) causa(s) de error.
- 3.8 Permitir enviar el trámite a la cola de pendientes a captación de imágenes.

**RF 4.** Permitir la visualización del expediente de una persona.

- 4.1 Mostrar datos personales.
- 4.2 Permitir imprimir datos personales.
- 4.3 Mostrar datos biométricos.
- 4.4 Permitir imprimir datos biométricos.
- 4.5 Mostrar historial de cambios de dirección.
- 4.6 Permitir imprimir cambios de dirección.
- 4.7 Mostrar historial de trámites.
- 4.8 Permitir imprimir historial de trámites.
- 4.9 Mostrar historial de documentos.
- 4.10 Permitir imprimir historial de documentos.
- 4.11 Mostrar poseedores.
- 4.12 Permitir imprimir poseedores.
- 4.13 Si la persona no posee nada de lo antes mencionado, mostrar un mensaje informativo.

**RF 5.** Imprimir Documento de Identidad Provisional (DIP).

- 5.1 Mostrar listado de nombres de personas pendientes a impresión del DIP.
- 5.2 Permitir seleccionar la persona.
- 5.3 Mostrar los datos del DIP.
- 5.4 Permitir seleccionar "Vista previa".
- 5.5 Mostrar los datos de acuerdo al formato del DIP.
- 5.6 Permitir imprimir DIP.
- 5.7 Si la impresión falla.
  - 5.7.1 Permitir reimprimir el DIP.
- 5.8 Si la impresión no falla.
  - 5.8.1 Mostrar el DIP al usuario para que revise sus datos.
  - 5.8.2 Si el DIP presenta errores.
    - 5.8.2.1 Permitir iniciar la Supervisión (Ver ERS Módulo Supervisión, **RF 1, RF 2,RF 3**)
  - 5.8.3 Si no presenta errores.
    - 5.8.3.1 Continuar con la impresión del próximo DIP.

### 2.3.4 Descripción de requisitos funcionales

**RF 1.** Permitir la supervisión de datos.

<b>Propósito</b>	Supervisar los datos de un trámite.	
<b>Roles</b>	Funcionario de Supervisión.	
<b>Precondiciones</b>	<ol style="list-style-type: none"> <li>1. El Funcionario de Supervisión debe estar autenticado en el sistema.</li> <li>2. El trámite que se desea supervisar debe haberse realizado en la unidad o encontrarse en curso.</li> </ol>	
<b>Entidades Tratadas</b>	<b>Entidad</b>	<b>Atributos</b>
	dPersona	<ul style="list-style-type: none"> <li>• Idpersona</li> <li>• idtipopersona</li> </ul>
	dNacional	<ul style="list-style-type: none"> <li>• idpersona</li> <li>• numeroidentidad</li> <li>• annoregistro</li> <li>• tomo</li> <li>• folio</li> <li>• idregistrocivil</li> <li>• primernombre</li> <li>• segundonombre</li> <li>• primerapellido</li> <li>• segundoapellido</li> <li>• nombremadre</li> <li>• nombrepadre</li> <li>• fechanacimiento</li> <li>• sexo</li> <li>• donante</li> <li>• talla</li> <li>• peso</li> <li>• fallecido</li> <li>• idcolorojos</li> <li>• idcolorpiel</li> <li>• idcolorcabello</li> </ul>

		<ul style="list-style-type: none"> <li>• idciudadania</li> <li>• iddatosopcionales</li> </ul>
	dTrámite	<ul style="list-style-type: none"> <li>• idpersona</li> <li>• idtipotrámite</li> <li>• idestadotramite</li> </ul>
	Error_datos	<ul style="list-style-type: none"> <li>• Omision de caracteres</li> <li>• Adicion de caracteres</li> <li>• Intercambio de datos</li> <li>• Intercambio de caracteres en un dato</li> <li>• Omisión de datos</li> </ul>
<b>Descripción</b>	<p>1.1 Mostrar pantalla con los datos de la persona que realiza el trámite.</p> <p>1.2 Si se detecta algún error en los datos</p> <ul style="list-style-type: none"> <li>a) Omisión de caracteres</li> <li>b) Adición de caracteres</li> <li>c) Intercambio de datos</li> <li>d) Intercambio de caracteres en un dato</li> <li>e) Omisión de datos</li> </ul> <p>1.2.2 Guardar la(s) causa(s) de error.</p> <p>1.3 Enviar trámite a la cola de Corrección de datos.</p> <p>1.4 Si no se detectan errores seguir con la supervisión.</p>	
<b>Validaciones</b>	<ol style="list-style-type: none"> <li>1. Validar la correctitud y completitud de los datos definidos en el Modelo de Datos.</li> <li>2. Los datos de la persona no pueden ser modificados.</li> </ol>	
<b>Postcondiciones</b>	<ol style="list-style-type: none"> <li>1. Se ha realizado la supervisión de datos del trámite indicado.</li> <li>2. El trámite fue enviado a la cola de captación de datos.</li> </ol>	

Prototipo

The screenshot shows a software window titled "Supervisar Trámite" with a blue header and standard window controls. The interface is divided into several sections:

- Navigation:** Tabs for "Datos", "Documentos", and "Imágenes".
- Form Fields:**
  - "Trámite:<Tipo de Trámite>" and "Estado del Trámite:<Estado>" (text labels).
  - "Foto": A placeholder icon of a person's head with a question mark.
  - "No. Identidad": Text box containing "86110621132".
  - "Nombre(s)": Text box containing "Gleydi".
  - "Primer Apellido": Text box containing "Claro".
  - "Segundo Apellido": Text box containing "Morgado".
  - "Sexo": Radio buttons for "M" (Male) and "F" (Female), with "F" selected.
  - Checkboxes for data manipulation: "Omisión de caracteres" (checked), "Adición de caracteres", "Intercambio de datos", "Omisión de datos", and "Intercambio de caracteres en un dato".
- Vínculos Personales:**
  - "Nombre Padre": Text box containing "Nombre Padre".
  - "Nombre Madre": Text box containing "Nombre Madre".
  - "Hijo(s)": Text box containing "Hijo(s)" with a small icon.
- Color de Piel:** Radio buttons for "Blanca" (selected), "Negra", and "Mestiza".
- Ciudadanía:** Dropdown menu showing "Ciudadanía".
- Ocupación:** Text box containing "Ocupación".
- Registro Civil:** Text box containing "Registro Civil".
- Tomo:** Text box containing "Tomo".
- Folio:** Text box containing "Folio".
- Año:** Text box containing "Año".

At the bottom, there are three buttons: "Enviar a Captación de Datos", "Aceptar", and "Cancelar".

Fig. 7 PIU Supervisar Datos

Para mejor comprensión de los requisitos funcionales y de las funciones que estos abarcan ver **Anexo 3: Descripción de los requisitos funcionales.**

### 2.4 Modelo de entidades conceptuales

Los modelos de entidades conceptuales ayudan a representar actividades de la vida diaria con un alto grado de abstracción. Dichos modelos están compuestos por objetos que no son más que contenedores de información y que a su vez tienen atributos que los representan e identifican. En la actividad de requerimiento se diseña el modelo de entidades conceptuales para lograr una aproximación a lo que debería ser el modelo de datos que se obtiene en el flujo de análisis y diseño.

Sirve para el análisis con el cliente de las entidades que deberán persistir una vez terminados los procesos.

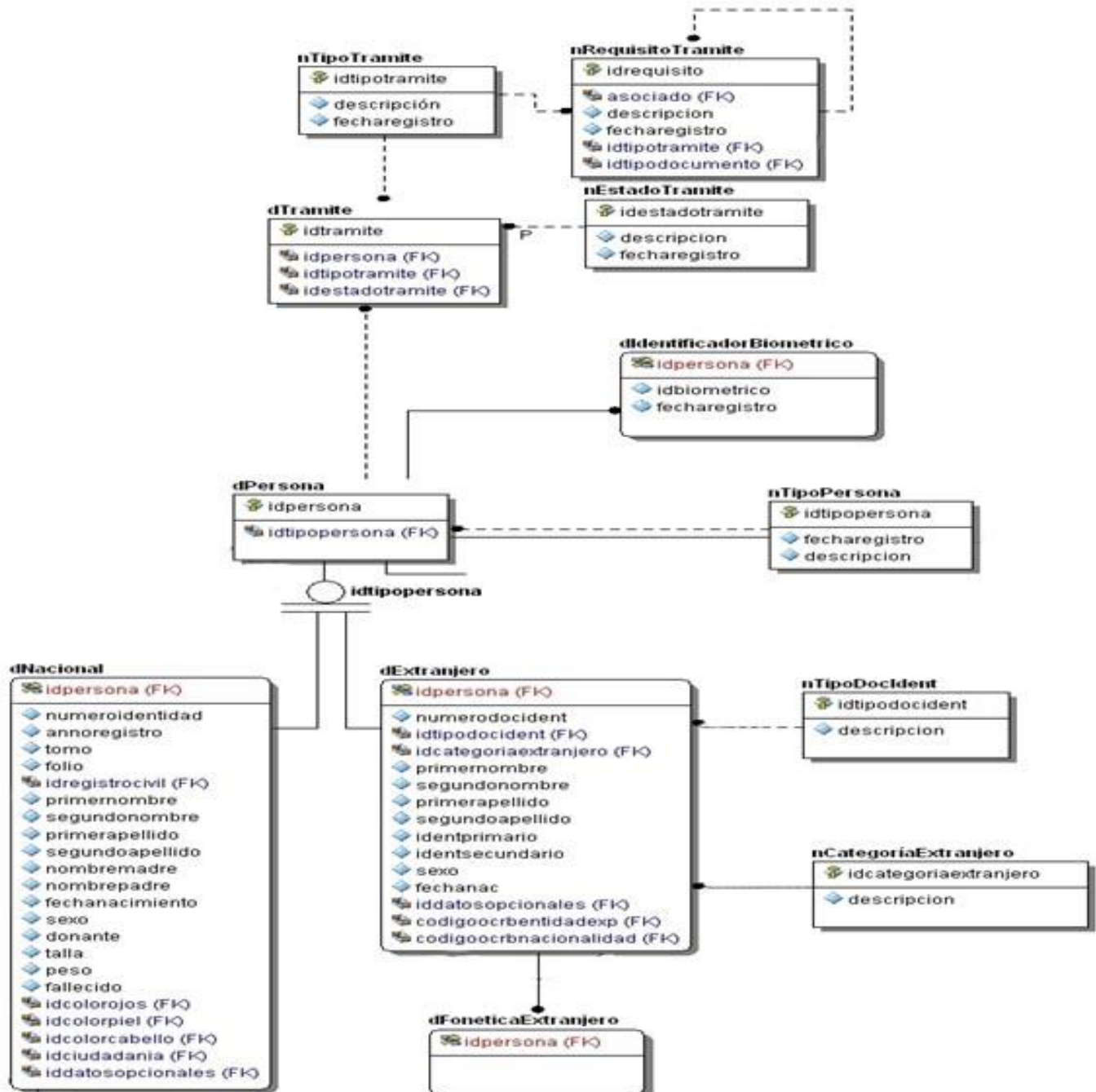


Fig. 8 Modelo de entidades conceptuales

### 2.4.1 Descripción de las entidades fundamentales

<b>Nombre de la entidad</b>	<i>dTramite</i>
<b>Definición</b>	<i>Entidad que representa los trámites asociados a las personas</i>
<b>Atributos</b>	<i>Idtrámite, ldestadotramite, ldtipotramite, ldpersona</i>

**Tabla 2** Descripción de la Entidad *dTramite*

<b>Nombre de la entidad</b>	<i>dPersona(dNacional)</i>
<b>Definición</b>	<i>Entidad que representa las personas cubanas que residen en Cuba</i>
<b>Atributos</b>	<i>ldpersona, numeroidentidad, annoregistro, tomo, folio, idregistrocivil, primernombre, segundonombre, primerapellido, segundoapellido, nombremadre, nombrepadre, fechanacimiento, sexo, donante, talla, peso, fallecido, idcolorojos, idcolorpiel, idcolorcabello, idciudadania, iddatosopcionales</i>

**Tabla 3** Descripción de la Entidad *dPersona (dNacional)*

<b>Nombre de la entidad</b>	<i>dDocumento</i>
<b>Definición</b>	<i>Entidad que registra los documentos</i>
<b>Atributos</b>	<i>lddocumento, ldtipodocumento, fecharegistro</i>

**Tabla 4** Descripción de la Entidad *dDocumento*

### 2.5 Especificación de requisitos no funcionales

Según la *IEEE Standard Glossary of Software Engineering Terminology* los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. Son importantes para que clientes y usuarios puedan valorar las características no funcionales del producto (23). A continuación se muestran algunos de los requisitos no funcionales definidos para el Módulo de Supervisión y Control.

#### Usabilidad

**RnF1.** El sistema podrá ser utilizado por cualquier usuario con las siguientes características:

- a) Conocimientos básicos relativos al uso de una computadora.
- b) Conocimientos básicos del sistema operativo Windows.
- c) Conocimientos sólidos relativos a los procesos de negocio acorde al rol que desempeñe.



**RnF2.** El sistema será distribuido en idioma español, aunque estará preparado para soporte multilinguaje.

**RnF3.** Los términos utilizados se establecerán acorde al negocio correspondiente para facilitar la comprensión de la herramienta de trabajo.

**RnF4.** El sistema poseerá estructura y diseño homogéneo en todas sus pantallas, que facilite la navegación.

- a) Menús laterales y desplegados que permitan el acceso rápido a la información.
- b) Menú de soporte que facilite el acceso a herramientas utilitarias, notificaciones del sistema y ayuda integrada.

### Seguridad

**RnF1.** El sistema estará disponible durante toda la semana en el horario laboral, según el área correspondiente.

**RnF2.** El sistema contará con diferentes niveles de acceso:

- a) De los usuarios: se realizará acorde con los roles autorizados para el uso de cada funcionalidad tanto a nivel de funciones de las aplicaciones como de información de las bases de datos
- b) De las oficinas: se registrarán con los roles que le corresponda acorde a las funciones que le competan.
- c) De las estaciones de trabajo: se registrarán las estaciones de trabajo de forma tal que cada una acceda a las aplicaciones y funcionalidades que le corresponda según su ubicación y función en la organización.

**RnF3.** El sistema registrará todas las acciones que se realizan.

- a) Se llevará el registro del tiempo de actividad y del lugar de acceso de cada usuario y estación de trabajo para cada acción.
- b) Se podrán realizar auditorías para la comprobación de las actividades realizadas en el sistema.

**RnF4.** La conexión a los servidores de bases de datos y de aplicaciones es requerida para el correcto funcionamiento.

**RnF5.** Las réplicas de información se realizarán con la menor latencia posible entre los nodos.

**RnF6.** El tiempo medio de reparación de fallas del software no atribuibles al hardware o a la conectividad se definirá a partir de los Acuerdos de Niveles de Servicios establecidos entre el MININT y la UCI.

**RnF7.** El sistema garantizará la mayor exactitud en los datos.

- a) Deberá tenerse en cuenta la calidad de la fuente inicial de los datos para la migración de los mismos a las nuevas bases de datos.
- b) Deberán realizarse validaciones y comprobaciones automáticas en todos los casos posibles para garantizar la consistencia de los datos.

### Interfaz

#### Interfaces de usuario.

**RnF1.** Interfaz accesible e intuitiva, el manejo de las funcionalidades del portal debe ser lo más intuitivo posible, de manera que sean muy claras las posibles acciones a llevar a cabo y la manera de hacerlas.

**RnF2.** Interfaz consistente con las pautas de diseño definidas para los sistemas del proyecto. Ver documento de Pautas de Interfaz del Sistema de Identificación, Migración y Extranjería.

**RnF3.** Internacionalización de la interfaz.

**RnF4.** Consistencia en la aplicación entre los distintos navegadores. Debe visualizarse y manejarse de la misma forma en los navegadores más comunes agrupados, a continuación según su motor/intérprete de *HTML*<sup>28</sup>, *CSS*<sup>29</sup> y JavaScript.

- a) Trident (Internet Explorer a partir de la versión 7).
- b) WebKit (Konqueror, Safari, Google Chrome)
- c) Gecko (Mozilla Firefox, Flock)
- d) Opera

---

<sup>28</sup> Siglas del inglés *HyperText Markup Language* (Lenguaje de Marcado de Hipertexto)

<sup>29</sup> Siglas del inglés *Cascading Style Sheets* (Hojas de estilo en cascada)

### Interfaces Hardware.

**RnF1.** Interfaz capacitada para conectarse a las diferentes impresoras que posea la organización para el puesto de Supervisión.

### Interfaces Software.

**RnF1.** Se requiere la librería (Dll) que permita hacer uso de las funcionalidades para el control de calidad de imágenes que provee el software Morpheus.

**RnF2.** Se requiere la librería (Dll) que permita hacer uso de las funcionalidades del componente de AFIS para la evaluación de calidad de huellas y la extracción de minucias.

**RnF3.** Se requieren las librerías (Dll) y componente que permitan la integración con el componente de *PKI*<sup>30</sup> para los elementos de seguridad en las comunicaciones y gestión de usuarios.

**RnF4.** Se requieren las librerías (Dll) y componente que permitan la integración con el componente *Device Grid Manager* para la gestión de los dispositivos de hardware con los que se interactúa en el sistema.

**RnF5.** Se requiere la librería para la actualización de las aplicaciones.

**RnF6.** Se requiere la librería (Dll) que permita hacer uso de la información de los Registros Operativos.

**RnF7.** Se requiere la librería de Oracle que permita conectarse al servidor de aplicaciones.

### Interfaces de Comunicación

**RnF1.** Los servicios web poseerán una interfaz que permita manejar un alto nivel de seguridad haciendo uso de *SSL*<sup>31</sup>.

**RnF2.** Se deben exponer servicios web que permitan la comunicación con los distintos sistemas que necesiten notificar u obtener información del sistema.

---

<sup>30</sup> Siglas del inglés *Public Key Infrastructure*(*Infraestructura de Llave Pública*)

<sup>31</sup> Siglas del inglés *Secure Socket Layer*(*Protocolo de Capa de Conexión Segura*)

### **Conclusiones**

- ❖ En el desarrollo de este capítulo se describieron las actividades actuales del negocio que enfrenta la entidad para la cual está destinado el sistema propuesto mediante documentación e información. Ha permitido el análisis de los procesos actuales del negocio y el arribo a la conclusión de la necesidad de independizar el Módulo de Supervisión y Control por la importancia dentro del flujo de tareas que tiene el mismo.
- ❖ Se expusieron los diagramas de proceso para mejor entendimiento de las tareas a cumplir por parte de los desarrolladores, describiendo con estos diagramas los caminos a seguir ante cualquier situación.
- ❖ Se muestra el catálogo de requisitos funcionales del módulo y, para mejor comprensión, la descripción de ellos permitiendo así el análisis profundo de las funcionalidades que los requisitos deben cumplimentar. Así como también se muestran los requerimientos no funcionales que definen comportamientos específicos del sistema.
- ❖ Se expone el modelo de entidades conceptuales describiéndose las entidades fundamentales y los atributos que las representan.

### Capítulo 3: Análisis y Diseño

#### Introducción

El propósito fundamental del presente capítulo es exponer los elementos utilizados en el proceso de construcción de software, guiado por la arquitectura y haciendo énfasis en sus características esenciales, así como la forma de dar cumplimiento a las funcionalidades identificadas. Provee además el conjunto de artefactos obtenidos a través de la aplicación de la metodología de desarrollo seleccionada, la interrelación entre cada uno de estos componentes y sus funciones específicas.

#### 3.1 Arquitectura de la solución

La arquitectura de software es el diseño de más alto nivel de la estructura de un sistema, donde se establecen los fundamentos para que analistas, diseñadores, programadores, etc. trabajen en una línea común que permita alcanzar los objetivos del sistema de información, cubriendo todas las necesidades.

#### Vista lógica del Módulo de Supervisión y Control

El Módulo de Supervisión y Control como parte del Sistema Único de Identificación Nacional combina cinco capas esenciales, donde cada nivel se especializa en una actividad específica para evitar la dependencia a una sola tecnología o aplicación. Ver **Fig. 9** *Componentes de los módulos de la aplicación.*

##### 3.1.1 Capas de los módulos de la aplicación

1. Capa de Presentación
2. Capa de Procesos y Servicios
3. Capa de Negocio
4. Capa de Acceso a Datos
5. Capa de Base de Datos

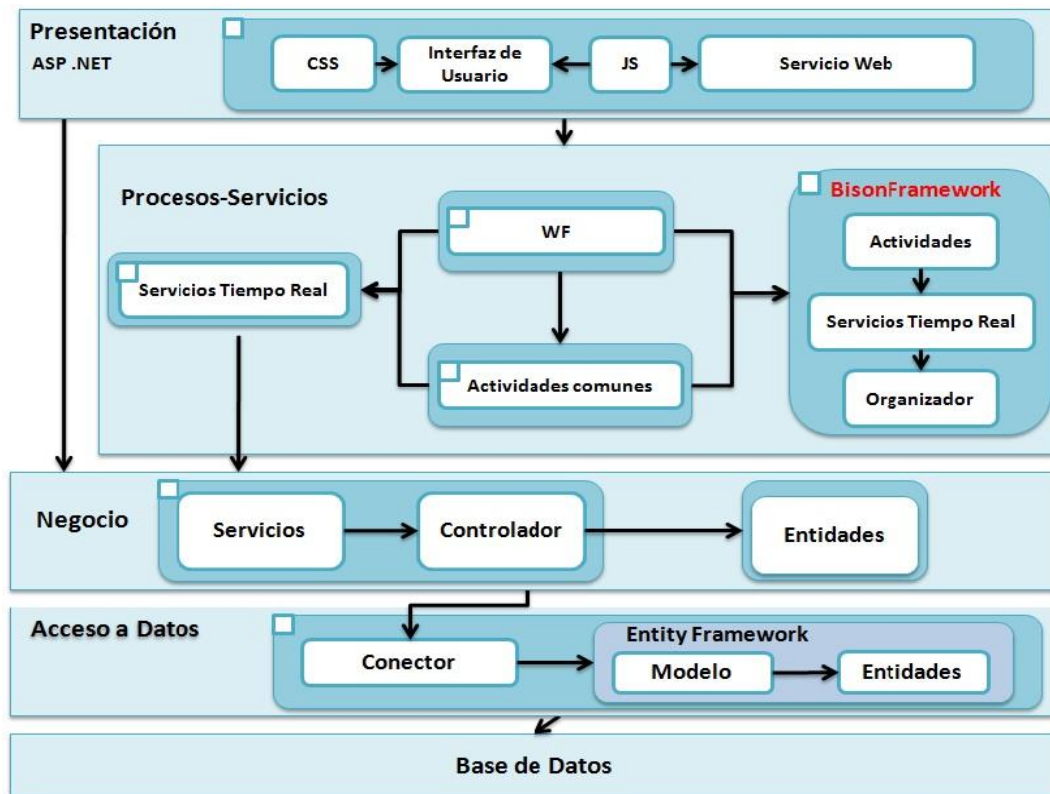


Fig. 9 Componentes de los módulos de la aplicación

### Presentación

Contiene los requerimientos y la lógica de la interfaz de usuario dentro de una capa separada que permite tanto su reutilización como su independencia de la lógica de negocio, permitiendo que se pueda modificar la interfaz sin que esto afecte al negocio y viceversa. En el sistema esta capa posee un grado de independencia mayor ya que el flujo de las interfaces está gestionado no por esta capa sino por el motor de procesos, logrando que un cambio en la secuencia de las actividades del proceso de negocio tenga cero consecuencias en la capa de presentación. Está compuesta por todas las interfaces de usuario y los componentes necesarios para su correcto funcionamiento. Estos elementos pueden ser ficheros JavaScript, CSS, servicios consumidos por Java Script, etc.

### Procesos y Servicios

Tiene contenida la lógica de los procesos de negocio representada por workflow, las actividades que por su nivel de reutilización o importancia lógica se encapsulan en una actividad propia del proyecto

y, los servicios de *Runtime*<sup>32</sup>, los cuales son los encargados de la interacción con los workflow definidos, estos a su vez interactúan con los servicios de la capa de negocio haciendo uso para ello de una fábrica de servicios que le da la instancia correcta de dicho servicio. Al mismo nivel se encuentra definida una fábrica para los servicios de *Runtime* que le permite a la capa de presentación interactuar con estos sin necesidad de que sepan la instancia concreta del servicio que utilizan. Todos estos elementos se encuentran vinculados directamente con el *Bison Framework*.

### ❖ **Bison Framework**

Es un *framework* para la orquestación de procesos de negocio con *Windows Workflow Foundation*. Su principal objetivo es proporcionar un componente que permita gestionar las instancias de workflow. Además encapsula un conjunto de actividades y servicios que le dan mayor dinamismo al desarrollo de sistemas centrado en la orquestación de procesos de negocio con workflow, específicamente para un ambiente web. Entre las ventajas que brinda *BisonFramework* se pueden mencionar:

- ❖ Proporciona una mayor aproximación a los usuarios de negocio.
- ❖ Brinda rapidez y flexibilidad para modelar y cambiar los procesos según las necesidades.
- ❖ Aporta escalabilidad o capacidad de crecer.
- ❖ Fortifica el puente creado por el workflow para la comunicación entre el analista y el desarrollador.
- ❖ Propone una arquitectura donde se encuentran bien definidas las capas de presentación y negocio.
- ❖ Posee actividades y servicios especializados en la orquestación de interfaces de usuario, que permiten definir su flujo de una manera gráfica dentro del workflow.

### **Negocio**

En esta capa se recogen todos los servicios necesarios para darle solución a los requisitos de negocio que no pueden ser satisfechos por el workflow.

Los servicios se encuentran definidos según el contexto en el que se desenvuelven. Tienen la responsabilidad de manejar todas las operaciones sobre una entidad de negocio en específico, así como todas las entidades que por conceptos de composición se encuentran relacionadas con esta.

---

<sup>32</sup> *Servicios en Tiempo Real*

Por cada entidad de negocio se crea un controlador y una interfaz que debe ser implementada por el acceso a datos que le dará soporte. Contiene entre sus proyectos:

- ❖ Entidades
- ❖ Servicios
- ❖ Controladores
- ❖ Interfaz Conector
- ❖ Interfaz Fábrica de los conectores

### **Acceso a Datos**

La capa de acceso a datos está directamente relacionada con los servicios definidos en el negocio. Para establecer esta relación hace uso de la interfaces de conectores y de la fábrica de conectores que define la capa de negocio. De esta manera es posible realizar cambios en esta capa sin que se vean afectadas las demás capas. Su principal función es realizar una implementación de las interfaces definidas en la capa de negocio y al mismo tiempo trabajar directamente con la fuentes de datos establecidas en la Base de Datos.

Por la importancia que posee el desarrollo de procedimientos y vistas dentro del negocio se ha separado la base de datos en una nueva capa, donde se concentran una pequeña parte de la lógica de funcionalidades dentro de la aplicación. La única capa encargada de interactuar con esta es *Capa de Acceso a Datos*.

### **Base de Datos**

Está constituida por todo el conjunto de tablas y procedimientos que permiten el almacenamiento de la información recolectada y procesada por los procesos. Por la importancia que posee el desarrollo de procedimientos y vistas dentro del negocio se ha separado la base de datos en una nueva capa, donde se concentran una pequeña parte de la lógica de funcionalidades dentro de la aplicación. La única capa encargada de interactuar con esta es *Capa de Acceso a Datos*.

Para mejor comprensión de los elementos que conforman la arquitectura del software ver el Documento de Arquitectura del proyecto Sistema de Identificación, Migración y Extranjería.



### 3.2 Patrones de diseño

El desarrollo de un sistema consiste, en la mayoría de las ocasiones, en darle solución a problemas muy complejos que ya alguien ha resuelto. Los patrones de diseño son una solución estándar para un problema común de programación, una técnica para flexibilizar el código haciéndolo satisfacer ciertos criterios entre otras.

#### 3.2.1 Patrones

Para el desarrollo del Módulo de Supervisión y Control se han tenido en cuenta un conjunto de patrones que permiten darle flexibilidad y no constituyen cambios grandes en el rendimiento del mismo, además se identificaron un conjunto de patrones específicos para el desarrollo de workflow que brindan claridad y fortaleza a los diseños de estos. A continuación se definen los patrones utilizados:

- ❖ **Encapsulación:** propone esconder algunos componentes, permitiendo sólo accesos estilizados al objeto. Se hace uso de este patrón en casi todas las clases que componen el sistema permitiendo que estas solo posean como elementos públicos aquellos que son exclusivamente necesarios.
- ❖ **Subclase:** propone heredar miembros por defecto de una superclase, seleccionando la implementación correcta a través de resoluciones sobre qué implementación debe ser ejecutada. Se puede encontrar este patrón con más fuerza en las entidades de negocio que por su conceptualización, las funciones y la información que almacenan pueden estar diferenciadas en cierta medida
- ❖ **Excepciones:** propone introducir estructuras de lenguaje para arrojar e interceptar excepciones. Se identificaron los diferentes tipos de errores a tratar dentro del sistema creando clases que permitan identificar cada tipo de error en el momento de ejecución.
- ❖ **Fábrica:** provee de una interfaz para crear familias de objetos relacionados o dependientes sin especificar los tipos concretos de clases. Su uso se encuentra centrado a la creación de los conectores correspondientes al acceso a datos que se esté utilizando, así como en la obtención de los servicios a utilizar.

❖ **Singleton:** se asegura que solo se pueda crear una instancia de la clase y ofrece un punto global de acceso a esta instancia. El uso de este patrón permite que los servicios puedan ser creados solo una vez.

❖ **GRASP:** el uso de este patrón está totalmente ligado a cada componente desarrollado en el sistema, donde cada uno de ellos posee solo las funcionalidades acorde a las particularidades que lo caracterizan.

### 3.2.2 Patrones Workflow

Los patrones van desde los más simples como el patrón secuencial hasta los complejos, por ejemplo, el patrón de sincronización. Los patrones de workflow pueden ser clasificados en las siguientes categorías.

❖ **Patrones de control de flujo básicos:** estos patrones están presentes en la mayoría de los lenguajes de workflow, y sirven para modelar procesos secuenciales, paralelos, o aquellos que incluyan alguna decisión.

❖ **Patrones de ramificación avanzada y sincronización:** estos patrones superan a los patrones de control de flujo básico al permitir tipos avanzados de bifurcación y sincronización.

❖ **Patrones estructurales:** permiten terminar un subproceso cuando ya no haya nada que hacer, o permiten definir ciclos de forma arbitraria.

❖ **Patrones que manejan múltiples instancias:** cuando se le da seguimiento a un caso, algunas veces es necesario que el proceso sea instanciado muchas veces.

## 3.3 Especificaciones de las clases

Para el acceso a los datos de la aplicación se hace necesario buscar apoyo en clases conectoras y gestoras que soporten el acceso directo a la base de datos. La definición de la arquitectura basada en capas ayuda a la recuperación de errores en tiempo y la asignación de responsabilidades a cada clase de forma que la información se maneje en orden y correctamente.

### 3.3.1 Clases Controladoras

Las clases controladoras utilizan a las clases conectoras utilizando los datos que devuelven y conformando las clases del negocio lo cual facilita el mantenimiento de las aplicaciones disminuyendo el esfuerzo de actualización en caso de ocurrir cambios en la fuente de datos, además, permite la realización de las pruebas a la aplicación mediante implementación de conectores de pruebas que no accedan a datos reales.

<b>Nombre</b>	<i>IPersonController</i>
<b>Descripción</b>	Clase que permite el manejo de la clase conectora.
<b>Métodos</b>	<b>Descripción</b>
<i>ByRange</i>	Método que devuelve un rango de una lista de personas a partir de un diccionario de datos dado.
<i>ByParams</i>	Método que devuelve una lista de personas a partir de un diccionario de datos dado.
<i>Count</i>	Método que devuelve la cantidad de personas existentes en la Base de Datos a partir de un diccionario de datos dado.
<i>CountTramiteByPerson</i>	Método que devuelve la cantidad de trámites en curso que posee una persona a partir del id de una persona dada.

**Tabla 5** Descripción de la Clase Controladora *IPersonController*

### 3.3.2 Clases Conectoras

Las clases conectoras son las que manejan directamente el acceso a la base de datos. Manipulan entidades de acceso a datos, y cuentan con la estructura de la lógica de acceso a datos, permitiendo una independencia total del gestor de base de datos a utilizar, el cambio en la bases de datos solo genera la actualización de estas clases.

<b>Nombre</b>	<i>IPersonConnector</i>
<b>Descripción</b>	Clase que permite la conexión del acceso a datos con el negocio
<b>Métodos</b>	<b>Descripción</b>
<i>ByRange</i>	Método que devuelve un rango de una lista de personas a partir de un diccionario de datos dado.
<i>ByParams</i>	Método que devuelve una lista de personas a partir de un diccionario de datos dado.
<i>Count</i>	Método que devuelve la cantidad de personas existentes en la Base de Datos a partir de un diccionario de datos dado.
<i>CountTramiteByPerson</i>	Método que devuelve la cantidad de trámites en curso que posee una persona a partir del id de una persona dada.

**Tabla 6** Descripción de la Clase Conectora *IPersonConnector*

### 3.3.3 Clases Entidades

Las clases entidades son las encargadas de modelar información que posee larga vida y que es a menudo persistente. Constituyen una representación de los datos guardados en la base de datos y

permite la presencia de los mismos a lo largo del ciclo de vida del workflow. Ofrecen soporte a los servicios y manejan actividades específicas. El Módulo de Supervisión y Control hace uso de la clase **DoRequest** definida por el Módulo de Recepción y que se utiliza para manejar los datos referentes a las entidades Trámite y Persona que son las que conforman esta clase. Además, son el sustento de información para la visualización del Expediente de la Persona.

<b>Nombre</b>	<i>DoRequest</i>		
<b>Descripción</b>	En ella se encuentran los datos generales de una solicitud.		
<b>Atributos</b>	<b>Tipo de Dato</b>	<b>Definición</b>	
<i>PTramite</i>	<i>List&lt;Tramite&gt;</i>	Lista que contiene los trámites de la persona	
<i>PPersona</i>	<i>Person</i>	Objeto que contiene los datos de la persona	

**Tabla 7** Descripción de la Clase Entidad *DoRequest*

<b>Nombre</b>	<i>Person</i>		
<b>Descripción</b>	Entidad que contiene los datos personales que tiene una persona		
<b>Atributos</b>	<b>Tipo de Dato</b>	<b>Definición</b>	
<i>IdentityNumber</i>	<i>string</i>	Carné de Identidad que identifica a la persona	
<i>RegisterYear</i>	<i>int</i>	Año en que se registra la persona en el Registro Civil	
<i>Tomo</i>	<i>int</i>	Número del libro de asiento	
<i>Folio</i>	<i>int</i>	Página donde está registrada la persona en el libro de asiento	
<i>FirstName</i>	<i>string</i>	Primer nombre de la persona	
<i>SecondName</i>	<i>string</i>	Segundo nombre de la persona	
<i>FirstSurname</i>	<i>string</i>	Primer apellido de la persona	
<i>SecondSurname</i>	<i>string</i>	Segundo apellido de la persona	
<i>MotherName</i>	<i>string</i>	Nombre de la Madre de la persona	

<i>FatherName</i>	<i>string</i>	Nombre del Padre de la persona
<i>BirthDate</i>	<i>DateTime</i>	Fecha de nacimiento de la persona
<i>Sex</i>	<i>char</i>	Sexo de la persona
<i>Donante</i>	<i>byte</i>	Carácter binario que indica si la persona es donante o no.
<i>Talla</i>	<i>int</i>	La talla de la persona
<i>Weigh</i>	<i>string</i>	El peso de la persona
<i>Dead</i>	<i>bool</i>	Carácter de tipo bool que indica si la persona es fallecida o no.
<i>MigrationCondition</i>	<i>string</i>	Condición migratoria de la persona
<i>Dma</i>	<i>string</i>	Día, Mes, Año de nacimiento
<b>Entidades adicionales que usa la persona</b>		
Entidad	Definición	
<i>Country</i>	Entidad que cuenta la información del país de la persona	
<i>EyeColor</i>	Entidad que cuenta con la información del color de ojos de la persona	
<i>SkinColor</i>	Entidad que cuenta con la información del color de piel de la persona	
<i>HearColor</i>	Entidad que cuenta con la información del color de pelo de la persona	
<i>Address</i>	Entidad que cuenta con las direcciones de la persona.	
<i>PersonType</i>	Entidad que cuenta con la información del tipo de persona	
<i>CivilRegister</i>	Entidad que cuenta con la información del registro civil de la persona	
<i>OptionalDate</i>	Entidad que cuenta con los datos opcionales de la persona	
<i>PersonImage</i>	Entidad que cuenta con las imágenes de la persona	

**Tabla 8** Descripción de la Clase Entidad Persona

### 3.3.4 Diagrama de clases del diseño

El diagrama de clases del diseño representa la interacción de los servicios con las entidades, los conectores a la base de datos y muestra cómo se comporta el flujo de informaciones en el sistema. Ver **Fig. 10** Diagrama de clases del diseño del requisito funcional Imprimir DIP.

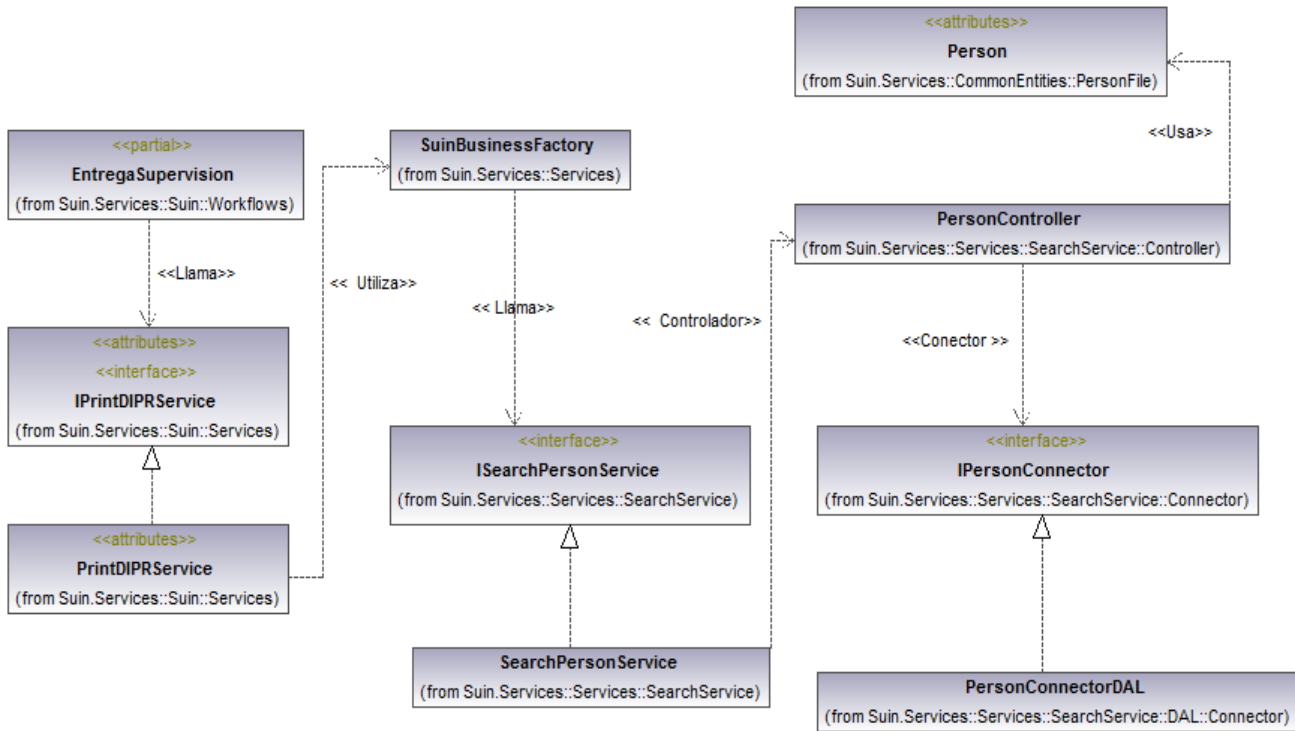


Fig. 10 Diagrama de clases del diseño del requisito funcional Imprimir DIP

### 3.4 Servicios del sistema

Un servicio no es más que una función, auto-contenida, que acepta llamada(s) y devuelve respuesta(s) mediante una interfaz bien definida que lo comunica con sistemas externos a él, independientemente del lenguaje de programación o tecnología con la que se trabaje. Los servicios pueden también ejecutar acciones discretas de trabajo como serían editar y procesar una transacción. Los servicios no dependen del estado de otras funciones o procesos, y los consumidores solo trabajarían en la orquestación de estos, secuenciar los servicios y proveer la lógica adicional para procesar datos. (24)

- ❖ Mejora en los tiempos de realización de cambios en procesos.
- ❖ Facilidad en modelos de negocios basados en colaboración con otros entes y colaboración con terceros.

- ❖ Facilidad para reemplazar elementos en nuestro servicio sin consecuencias en el proceso de negocio.

En el módulo de Supervisión y Control los servicios han sido desglosados, según las funciones específicas en las que trabaje, en dos grupos: Servicios de Runtime y Servicios de Negocio.

**Servicios de Runtime:** Trabajan en la captura y manejo de eventos con el usuario, es quien controla y maneja el intercambio de información con la capa de presentación. Tiene la responsabilidad de gestionar los eventos que comunican el sistema con el flujo de procesos (Capa de Procesos y Servicios).

**Servicios de Negocio:** Comunican el flujo de procesos con las capas inferiores, gestiona junto a las clases gestoras, controladoras y conectoras la información contenida en la base de datos.

A continuación se muestran algunos ejemplos de los servicios de negocio que utiliza el Módulo de Supervisión y Control. Para mejor comprensión ver **Anexo 5: Descripción de los Servicios del Runtime**.

### 3.4.1 Descripción de los servicios del sistema

#### Servicios del Runtime

<b>Nombre del Servicio</b>	<i>ICaptarDatosDocRService</i>
<b>Descripción</b>	<i>Controla la información correspondiente a los documentos captados</i>
<b>Métodos</b>	<i>LlenarAutorizo(Guid instanceld, Autorizo data )</i> <i>LlenarCertificado(Guid instanceld, Certificado data)</i>
<b>Argumentos</b>	<i>EventHandler&lt;CertificadoArgs&gt; OnCertificado;</i> <i>EventHandler&lt;AutorizoArgs&gt; OnAutorizo;</i>

**Tabla 9** Servicio *ICaptarDatosDocRService*

<b>Nombre del Servicio</b>	<i>IChangesPersonSuperviceRService</i>
<b>Descripción</b>	<i>Gestiona los cambios producidos a la persona durante el proceso de Supervisión y Corrección de los Datos.</i>
<b>Métodos</b>	

	<i>EventCallPerson(Guid instancelId, Person persona)</i>
<b>Argumentos</b>	<i>EventHandler&lt;DataPerson&gt; OnEventPerson;</i>

**Tabla 10** *IChangesPersonSuperviceRService*

<b>Nombre del Servicio</b>	<i>IPrintDIPRService</i>
<b>Descripción</b>	<i>Maneja los datos de la persona en la impresión del documento identidad(DIP)</i>
<b>Métodos</b>	<i>Print(Guid instancelId, Person persona)</i>
<b>Argumentos</b>	<i>EventHandler&lt;DataPerson&gt; OnReceiveDIP;</i>

**Tabla 11** *Servicio IPrintDIPRService*

### Servicios de Negocio

<b>Nombre del Servicio</b>	<i>ISearchPersonService</i>
<b>Descripción</b>	<i>Busca las personas que cumplan determinadas condiciones, es usado en la confección del Expediente de la Persona</i>
<b>Métodos</b>	<i>ByParams(Dictionary&lt;Object, Object&gt; parameters)</i> <i>int Count(Dictionary&lt;Object, Object&gt; parameters);</i> <i>Person GetPersonbyId(string personId);</i>

**Tabla 12** *Servicio ISearchPersonService*

### 3.5 Diseño de Workflow

La tecnología Workflow Foundation automatiza una secuencia de acciones, actividades o tareas utilizadas para la ejecución de cierto proceso, captura la interacción entre entidades del mundo real, incluyendo el seguimiento del estado de cada una de ellas y las herramientas necesarias para gestionarlo.

Para manejar cada una de las entidades con las que interactúa el workflow se necesitan actividades que manipulen la información contenida dentro del proceso, las que, entre otras funciones aportan herramientas similares a las usadas en distintos lenguajes y que cumplen tareas equivalentes. Algunas de estas estructuras de control del flujo son:



1. Condicionales (*IfElseActivity, Clases Policy Activity*).
2. Bucles While – Until (*Clases WhileActivity, ReplicatorActivity*)
3. Disparo y Captura de excepciones (*Clases ThrowActivity, FaultHandlerActivity*)
4. Manejo de Eventos (*HandleExternalEventActivity, EventDrivenActivity*)
5. Ámbito de Ejecución (*Clase SincronizationScopeActivity*) (11)

Las actividades son las encargadas de manipular y procesar la información, así como la comunicación del workflow con el mundo externo al flujo de procesos, son bloques con los que se construyen los flujos de trabajo. Las Actividades pueden ser simples o compuestas, las que son creadas por el propio desarrollador al heredar de la clase *Activity* o combinando varias actividades simples.

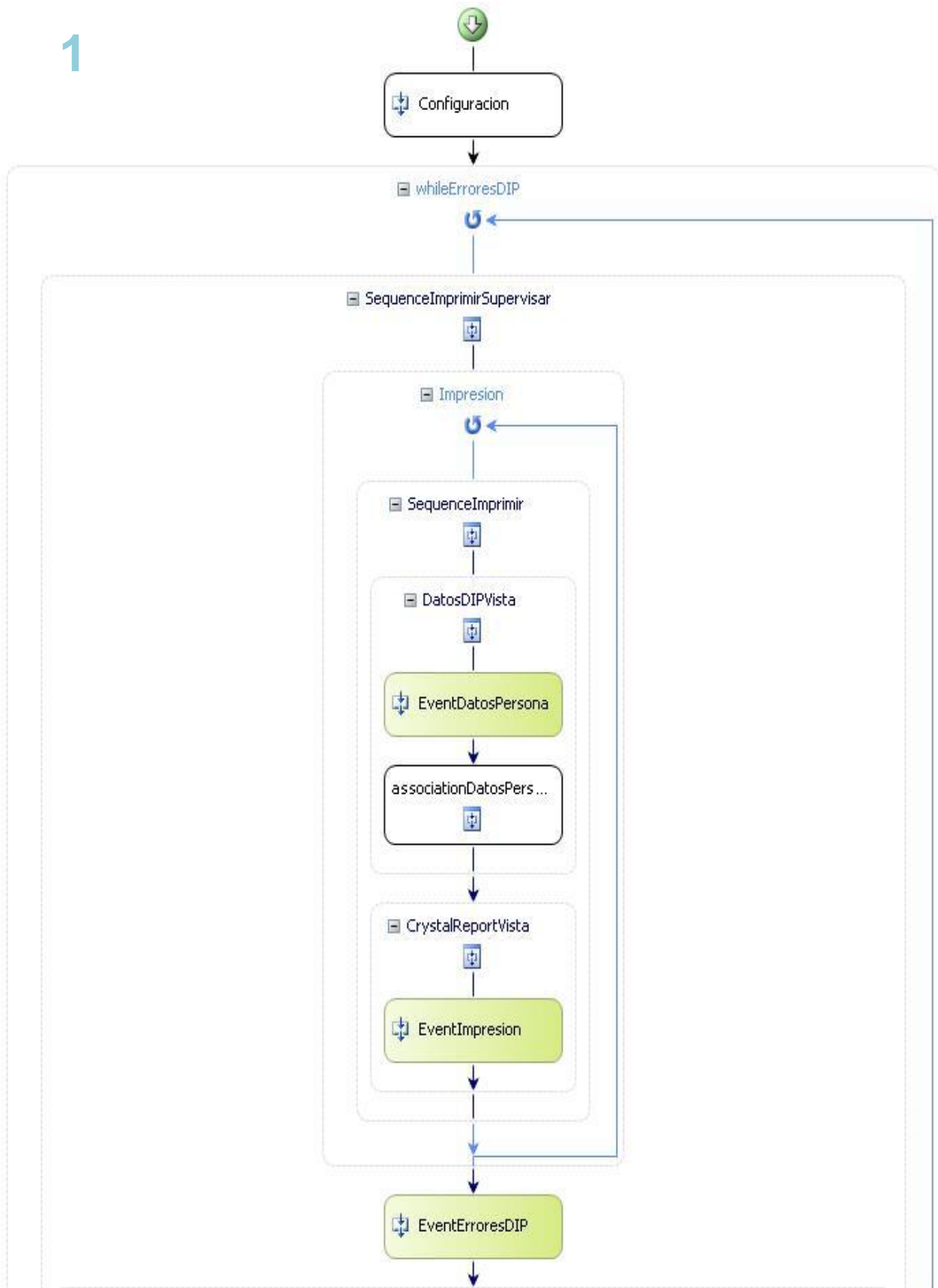
### ***Transformación del modelo de proceso mejorado a Workflow Foundation***

El Módulo de Supervisión y Control inicia su flujo de eventos cuando el funcionario de la oficina de CIRP necesita la verificación de los datos de un trámite que presenta irregularidades y requiere la validez de este. El otro flujo de eventos que pudiera iniciarlo es la impresión del Documento de Identidad Provisional (DIP).

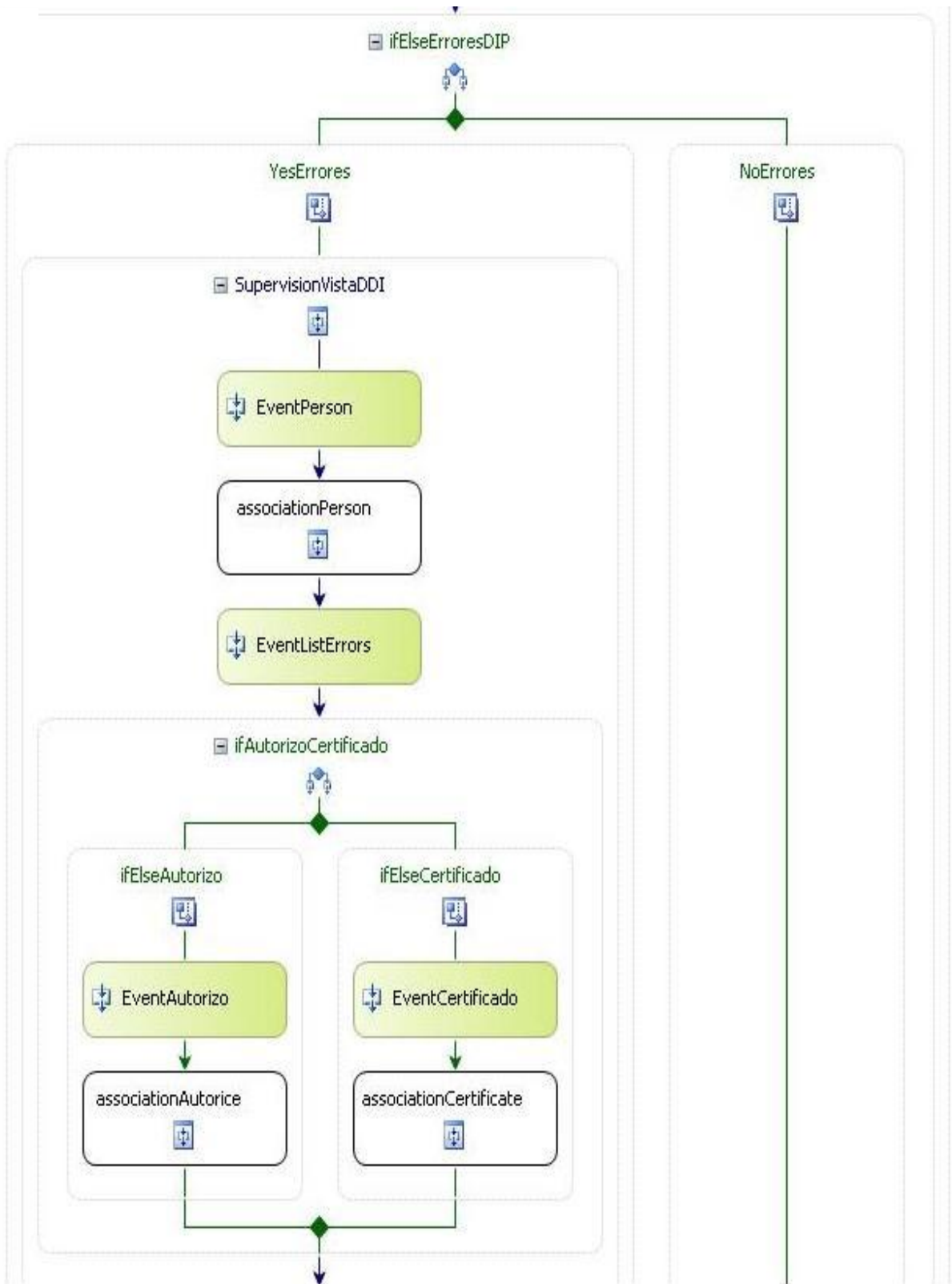
En la transformación del proceso de Negocio hacia la modelación de workflow cada acción del proceso de mejora pudiera desencadenar varias actividades, dependiendo de la complejidad del negocio en el que se esté trabajando y la cantidad de entidades que se estén manejando. A pesar de parecer esto un poco más trabajoso que una implementación con otra tecnología, daría al proceso de desarrollo una mayor independencia ante cualquier cambio en la lógica de Negocio, donde sería muy sencilla su reestructuración. Ver **Fig. 11** *Diseño de Workflow para el Módulo de Supervisión y Control*.

Para llevar a cabo el diseño del workflow se busca apoyo en las actividades que guían el proceso. Este apoyo lo ofrecen las actividades de Workflow Foundation que cumplen cada una con funciones específicas. Para mejor entendimiento ver **Anexo 4: Especificación de las actividades utilizadas en el diseño del workflow**.

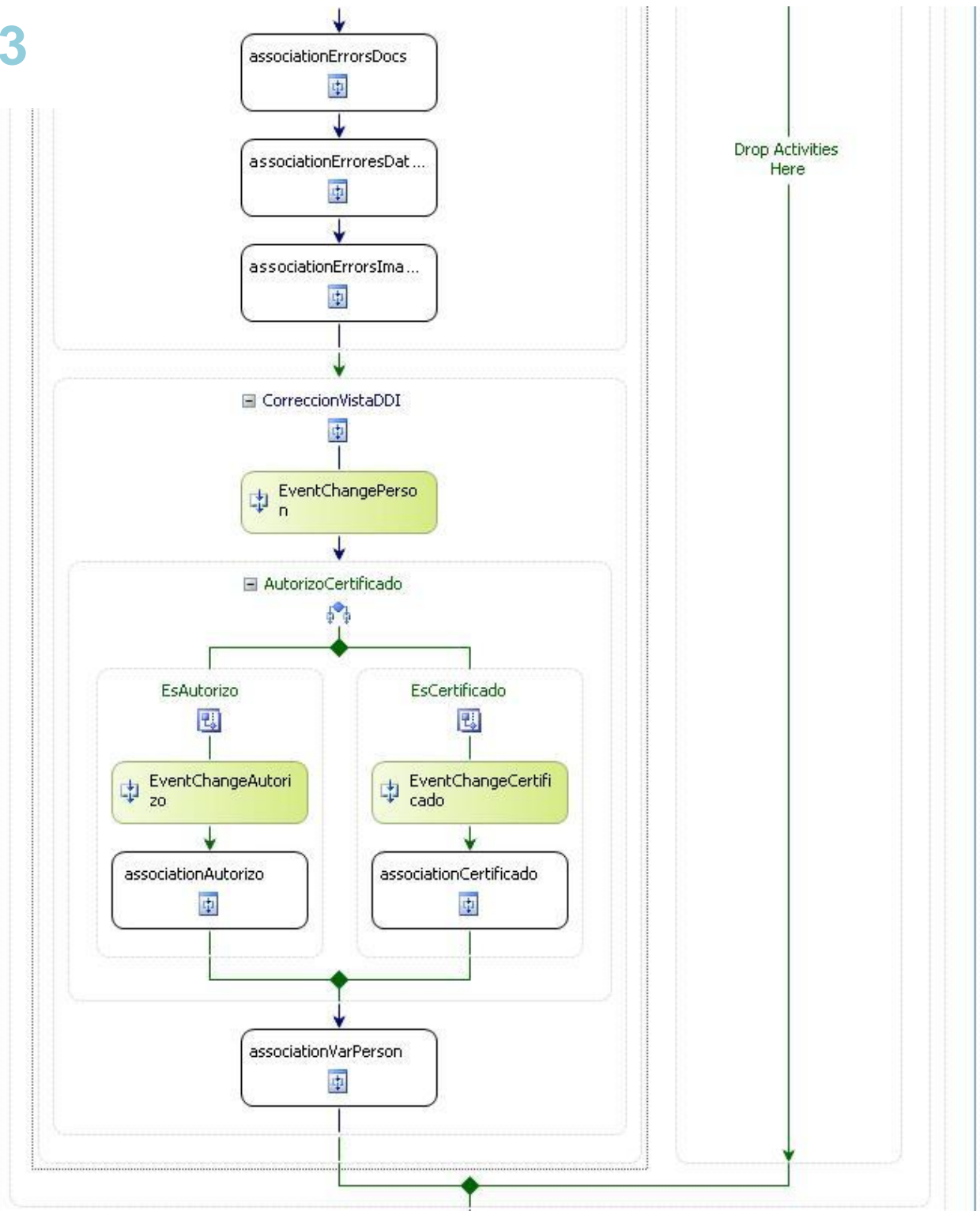
1

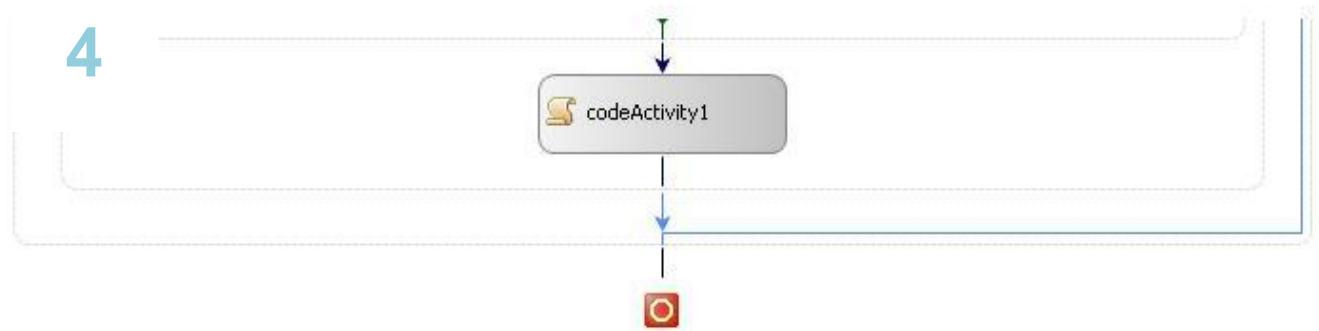


2



3





*Fig. 11 Diseño de Workflow para el Módulo de Supervisión y Control*

### 3.6 Modelo de datos

Un modelo de datos es un lenguaje utilizado para la descripción de una base de datos. Por lo general, permite describir los elementos que intervienen en una realidad o problema dado y la forma en que se relacionan dichos elementos entre sí. (25)

En el Capítulo 2 se presentó un modelo de entidades conceptuales con el propósito de realizar las validaciones de los requisitos con el cliente. En el presente capítulo se puede observar el cambio de este modelo y se definen de una vez, las entidades que persisten en la base de datos y los atributos que las identifican.





3.6.1 Descripción de las entidades fundamentales

<b>Nombre Tabla</b>	<i>dPersona</i>
<b>Descripción</b>	<i>Entidad que representa la generalización de la Persona.</i>
<b>Atributos</b>	<i>Idpersona, idtipopersona</i>

**Tabla 13** Entidad Persona

<b>Nombre Tabla</b>	<i>dNacional</i>
<b>Descripción</b>	<i>Entidad que representa las personas que viven en Cuba.</i>
<b>Atributos</b>	<i>idpersona, numeroidentidad, primernombre, segundonombre, primerapellido, segundoapellido, fechanacimiento, sexo, fallecido, tomo, folio, annoregistro, fechaactualizacion, idregistrocivil, idpais, idprovincianacimiento, idmunicipionacimiento</i>

**Tabla 14** Entidad Persona Nacional

<b>Nombre Tabla</b>	<i>dPersonaOtrosDatos</i>
<b>Descripción</b>	<i>Entidad que representa los datos opcionales de las personas.</i>
<b>Atributos</b>	<i>idpersona, nombrepadre, nombremadre, sennasparticulares, peso, talla, correo, donante, telefonparticular, telefonocelular, telefonolaboral, oficio, profesion, centrolaboral, idnivelcultural, idcolorpiel, idcolorojos, idcolorcabello, idcondicionresidencia</i>

**Tabla 15** Entidad PersonaOtrosDatos

<b>Nombre Tabla</b>	<i>dPersonalimagen</i>
<b>Descripción</b>	<i>Entidad que representa las imágenes de las personas.</i>
<b>Atributos</b>	<i>idpersonaimagen, imagen, fecharegistro, idpersona, idtipoimagen, idinfacial</i>

**Tabla 16** Entidad Personalimagen

Para tener mejor dominio de las entidades que intervienen en el modelo de datos ver **Anexo 6: Descripción de las entidades fundamentales.**

### 3.7 Conclusiones

- ❖ El estudio del capítulo permite entender la estructura organizacional de la arquitectura de la aplicación así como los módulos que lo conforman y, que unidos, integran el sistema con sus capacidades.
- ❖ Se describen las clases en las cuales se soporta el desarrollo del sistema propuesto especificando su tipo (entidades, controladoras y gestoras) y funciones específicas dentro de la aplicación. Se detallan los servicios de negocio y de *runtime* que sirven de sustento al cumplimiento de las exigencias planteadas por el cliente.
- ❖ Se muestra el diseño del workflow obtenido a partir del diagrama de mejora de procesos donde cada actividad da cumplimiento a requerimientos planteados en el capítulo Características del Sistema.
- ❖ Se obtiene el modelo de datos real de la aplicación observándose en este los cambios a partir del modelo de entidades conceptuales mostrado en el Capítulo 2.



### Capítulo 4: Implementación y Prueba

#### Introducción

En el presente capítulo se muestra una representación física de los módulos de la aplicación, como se implementó esta enfocándose en el diagrama de despliegue y el diagrama de componentes, conformándose de esta forma un modelo de implementación que permita describir los componentes a construir y su organización y dependencia entre nodos físicos en los que la aplicación funcionará. Se cumple además la tarea de validación de las especificaciones del cliente a través de las Pruebas Unitarias lo que permite verificar el cumplimiento de los requisitos definidos y garantizar la calidad del software desarrollado.

#### 4.1 Estándares de codificación y tratamientos de errores

Los estándares de codificación se definen por el equipo de desarrollo para lograr generalización en la programación del software. La generalización de aspectos tan simples como el trato de las mayúsculas, ayuda a eliminar conflictos de funcionalidades implementadas con nombres iguales y guían de forma clara el proceso de desarrollo. A continuación se muestran algunos de los ejemplos de estándares que se tuvieron en cuenta para el desarrollo del Módulo de Supervisión y Control. Para mejor información sobre el tema ver el documento Estándares de codificación elaborado por el Grupo de desarrollo del Sistema de Identificación, Migración y Extranjería.

##### 4.1.1 Estilos para la capitalización

###### Pascal

La primera letra en el identificador y la primera letra de cada subsiguiente palabra concatenada se capitalizan. Puede utilizar los identificadores de Pascal case en caso de tres o más caracteres. Por ejemplo:

*BackColor*

###### Camello

La primera letra en el identificador está en minúscula y la primera letra de cada subsiguiente palabra concatenada es mayúscula. Por ejemplo:

*backColor*

### Mayúscula

Todas las letras en el identificador se capitalizan. Esta convención se utilizará sólo para los identificadores que constan de dos o menos letras. Por ejemplo:

```
System.IO
```

```
System.Web.UI
```

#### 4.1.2 Sensibilidad a mayúsculas

- ❖ No se deberá utilizar nombres o identificadores que requieran ser case *sensitivity*<sup>33</sup>.
- ❖ No se deberá crear dos *namespaces*<sup>34</sup> que se diferencien solo en el uso de las mayúsculas.
- ❖ No crear funciones con nombres de parámetros que se diferencian solo en el uso de la mayúscula.
- ❖ No se deberá crear namespaces con nombres de clases que se diferencien solo en el uso de las mayúsculas.
- ❖ No crear clases con propiedades que se diferencien solo en el uso de las mayúsculas.
- ❖ No crear clases con métodos que se diferencien solo en el uso de las mayúsculas.

#### 4.1.3 Tratamiento de errores

El diseño de un sistema no solo debe tener en cuenta lo que debe ocurrir, sino que además debe realizar un análisis profundo de las diferentes situaciones que se puedan presentar y que constituyen algún tipo de violación o de situación en particular que provocaría un error dentro del sistema.

Las excepciones como se menciona en el párrafo anterior son parte del diseño aun cuando la gran mayoría de ellas son identificadas en el proceso de implementación, no obstante es necesario definir con antelación un mecanismo efectivo para su tratamiento. En el caso del sistema a desarrollar es necesario llevar una traza de todas aquellas excepciones que se lancen, y en muchos casos es necesario que el usuario que está interactuando con la aplicación en ese momento especifique alguna información que tribute a la futura validación o corrección de la excepción lanzada.

Para el tratamiento de excepciones se definirá una interfaz base que permite dejar una traza clara de todas las excepciones lanzadas dentro del sistema. Además se pueden considerar las excepciones como parte del flujo de negocio en muchos procesos por lo cual es necesario lograr identificar cada tipo

<sup>33</sup> *Sensible a las mayúsculas o minúsculas*

<sup>34</sup> *De la definición en inglés de espacio de nombres*

de excepción para darle un tratamiento en particular a cada una, para ello se debe definir una clase en particular para aquellas excepciones que sean de interés para el negocio.

#### 4.2 Diagrama de componentes

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos del software que entran en la fabricación de aplicaciones informáticas. (26)

El diagrama de componentes del Módulo de Supervisión y Control (Ver **Fig. 13 Diagrama de Componentes del Módulo de Supervisión y Control**) se ha elaborado de acuerdo a la arquitectura que da soporte al desarrollo de la aplicación. Se ha estructurado de forma que se pueda apreciar la relación de cada uno de los componentes con los restantes y que le dan sustento a las funcionalidades del software. Se muestran los artefactos que construyen los servicios web, las entidades, elementos de presentación y acceso a datos y cómo se referencian entre ellos.

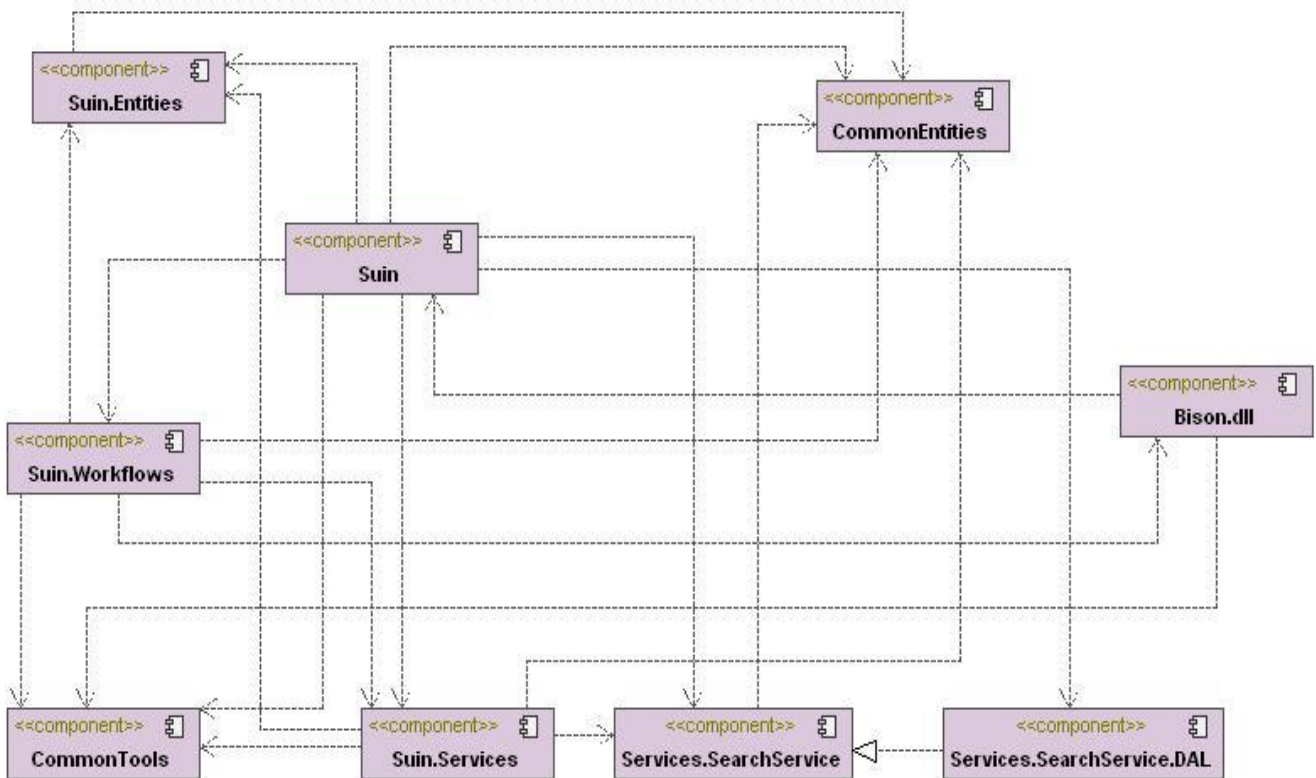
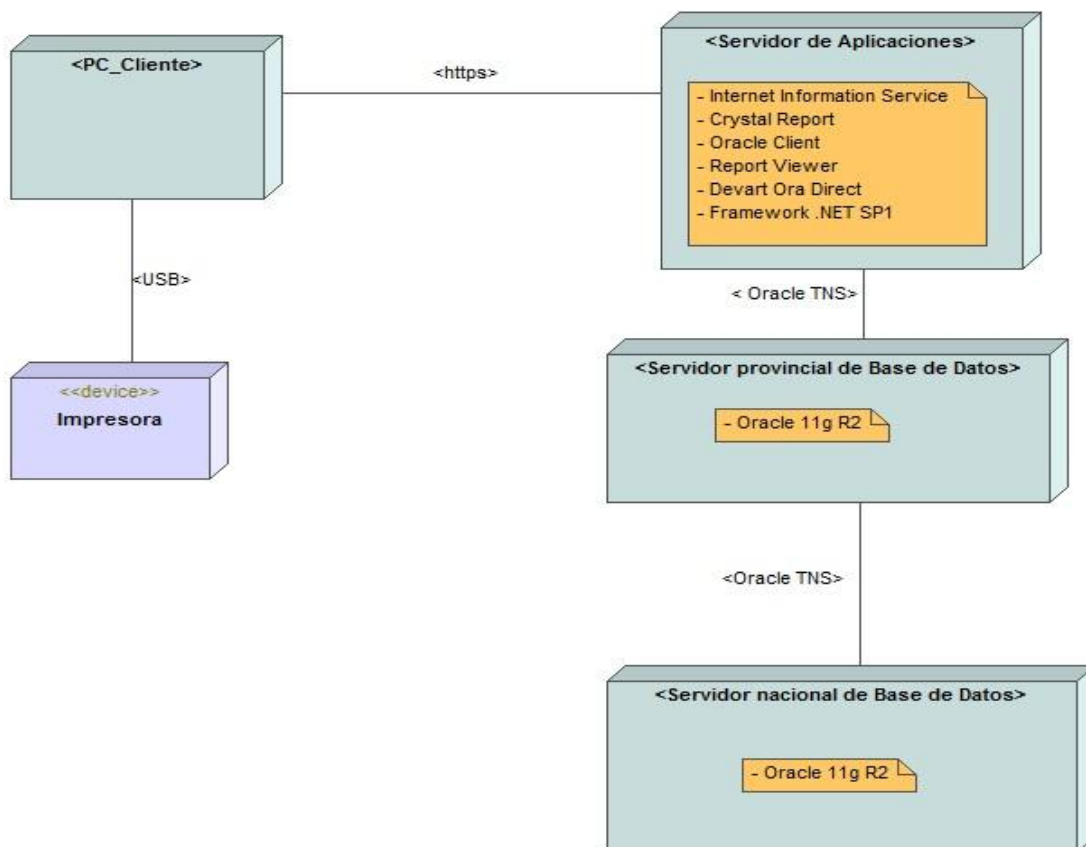


Fig. 13 Diagrama de Componentes del Módulo de Supervisión y Control

### 4.3 Diagrama de despliegue

Los Diagramas de Despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación. Un nodo es un recurso de ejecución tal como un computador, un dispositivo o memoria. (27)



**Fig. 14** Diagrama de despliegue del Módulo de Supervisión y Control

El diagrama de despliegue del Módulo de Supervisión y Control (Ver **Fig. 14** Diagrama de despliegue del Módulo de Supervisión y Control) muestra la distribución del sistema en función de los nodos que lo forman. La máquina cliente se conecta al servicio web quien es el que se encarga de ejecutar las funcionalidades y de ahí a la base datos para la gestión de datos en caso de que sea necesario. El dispositivo de impresión se debe incluir para la impresión del Documento de Identidad Provisional.

#### 4.4 Interfaces del sistema

Las interfaces del sistema son la cara de cada aplicación. Una interfaz cómoda, amigable y sencilla de entender por el usuario garantiza mejores procesos y guía el buen funcionamiento de la empresa donde será implantado. Para el desarrollo del Módulo de Supervisión y Control se tuvieron en cuenta los aspectos necesarios para lograr un diseño favorable al consumidor final del software. A continuación se muestran algunas de esas interfaces.

**Gsuin** SISTEMA ÚNICO DE IDENTIFICACIÓN NACIONAL

NOMBRE DEL SUBSISTEMA: Recepción Bienvenido: Adrian torres Graña | Salir | Ayuda

Administración  
 Recepción  
 Captura de Datos  
 Captura de Imágenes  
 Entrega  
 Supervisión  
 Supervisar Tramites  
 Reportes

**Recepción**

Imágenes Datos Documentos

**Datos generales**

Carné de identidad: 87090806582

Primer nombre: René Segundo nombre: Abilio Primer apellido: Piñeiro Segundo apellido: Hernández

Madre: Maruchi Padre: Emilio Sexo: M

**Datos del nacimiento**

Tomo: 238 Folio: 484 Año de registro: 1988 Registro civil: Arroyo Naranjo

**Datos secundarios**

Peso(kg): 75 Talla(cm): 180 Donante: Sí

Color piel: Blanca Color ojos: Negros Color pelo: Negro

**Datos adicionales**

Ocupación: Estudiante Dirección: Libertad s/n e/ Constanza y Lindero

Aceptar Cancelar

Fig. 15 Interfaz Supervisar datos

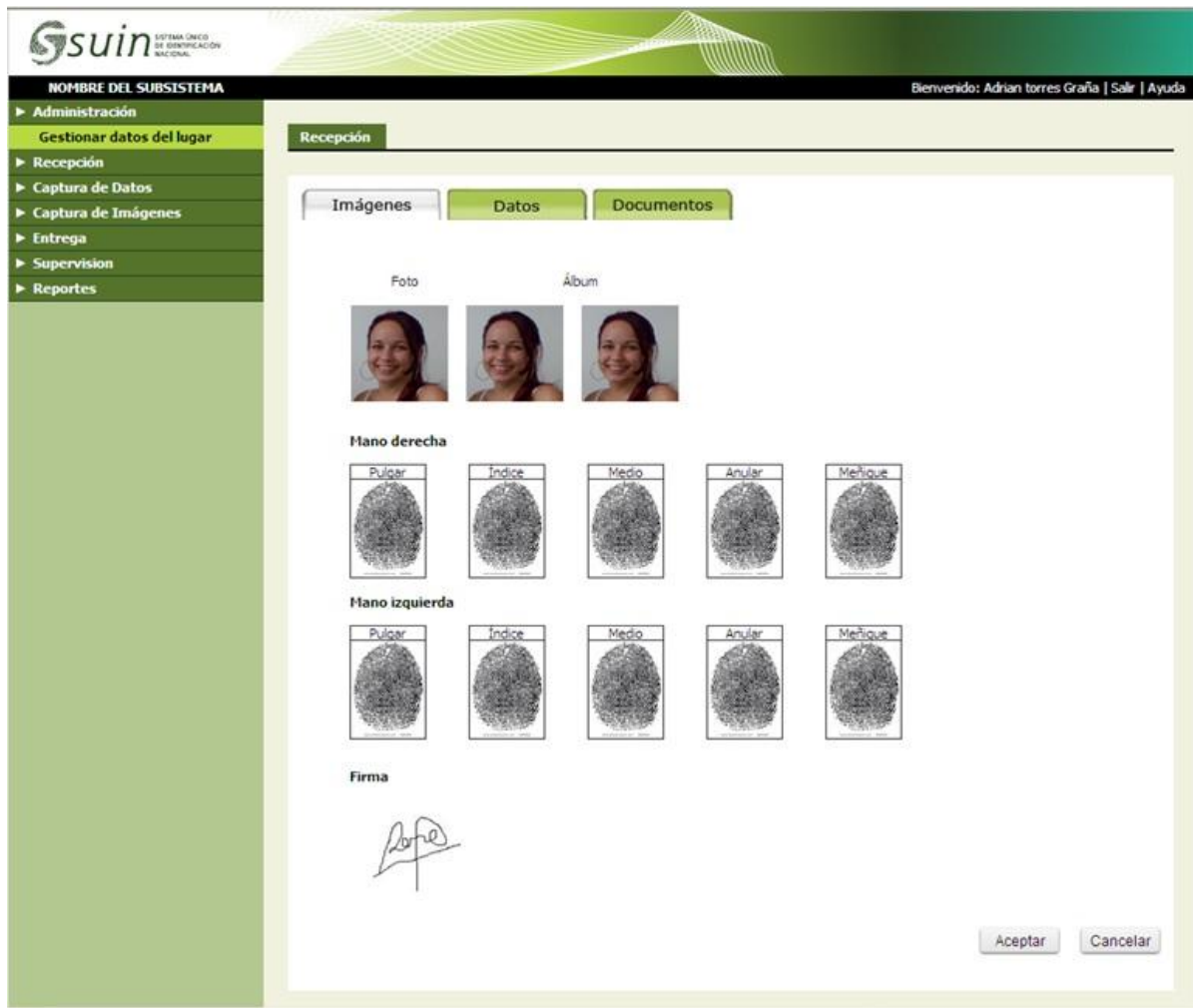


Fig. 16 Interfaz Supervisor imágenes

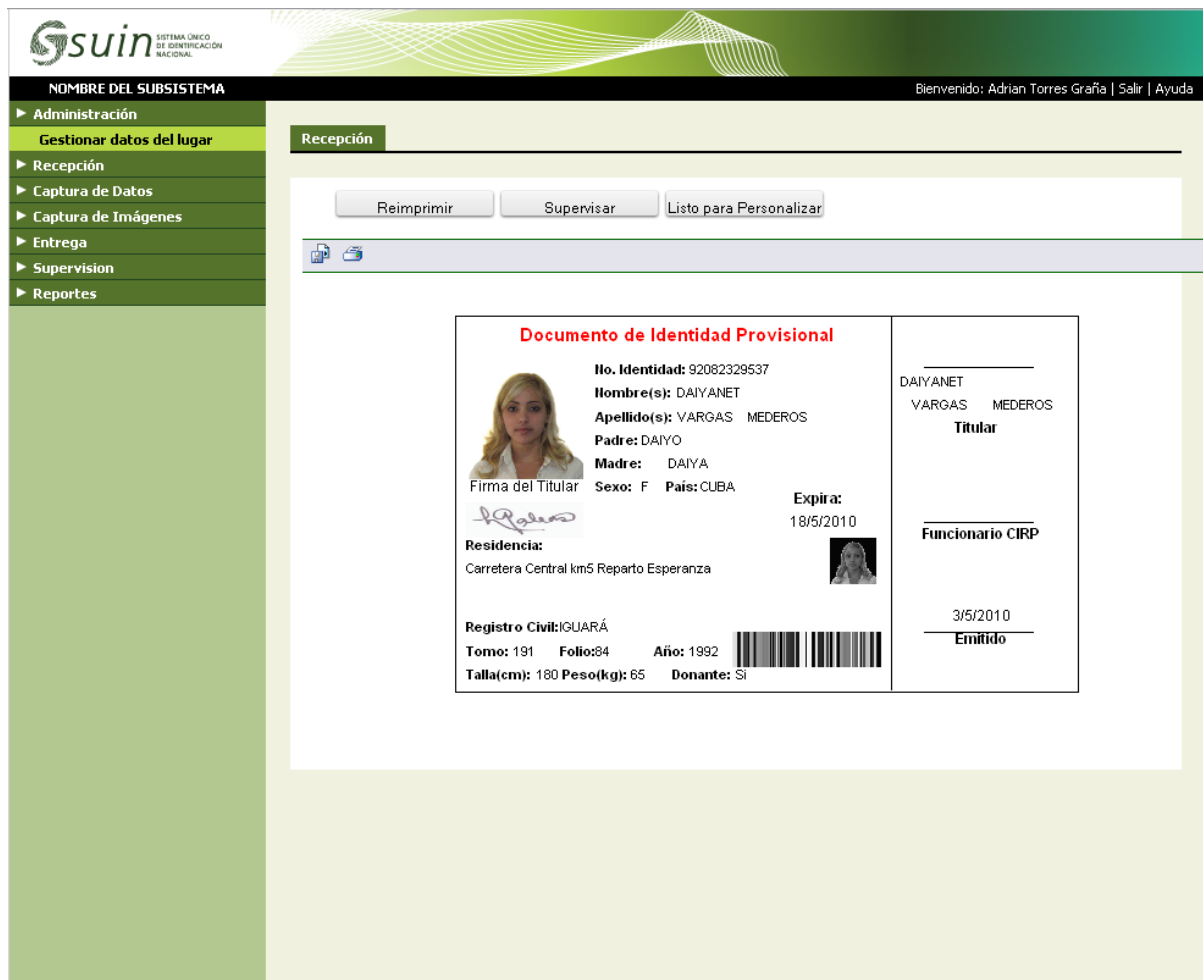


Fig. 17 Interfaz Imprimir DIP

#### 4.5 Pruebas

Las pruebas son una actividad en la cual un sistema o componente es ejecutado bajo unas condiciones o requisitos especificados, los resultados son observados y registrados, y se hace una evaluación del sistema o componente. La prueba de software es un elemento crítico para la garantía de la calidad del sistema y representa una revisión final de las especificaciones del diseño y de la codificación.

*Visual Studio Team System Test* incluye un conjunto de herramientas de prueba perfectamente integradas con Visual Studio, que no sólo funcionan en su propio marco de pruebas, sino también en un marco más amplio de herramientas de ciclo de vida del software. *Test Edition* permite crear, administrar, editar y ejecutar pruebas, así como obtener y almacenar los resultados de las pruebas. Incluye varios tipos de pruebas, como manuales, unitarias, de web y de carga. Además, también se incluye la medición de cobertura de código. (28)

#### 4.5.1 Diseño de Casos de Prueba

Los diseños de casos de prueba se llevan a cabo con el objetivo de medir los flujos de eventos de los requisitos y obtener a partir de estas los fundamentos que sustenten la calidad del software. A continuación (Ver **Tabla 17** Diseño de Caso de Prueba para el requisito Supervisar datos) se muestra el diseño de casos de prueba para el requisito funcional Supervisar datos.

##### CPR 1: Permitir la supervisión de datos

Condiciones de ejecución:

1. El Funcionario de Supervisión debe estar autenticado en el sistema.
2. El trámite que se desea supervisar debe haberse realizado en la unidad o encontrarse en curso.

Nombre del Flujo	Escenarios del flujo	Descripción de la funcionalidad	Flujo central
1. Supervisión de datos.	1.1: Permitir la supervisión de datos.	Permite supervisar los datos de un trámite.	<ol style="list-style-type: none"> <li>1. Mostrar pantalla con los datos de la persona que realiza el trámite.</li> <li>2. Si no se detectan errores seguir con la supervisión.</li> </ol>
3.3 Error en los datos	3.3. a Enviar trámite a captación de datos.	Permite guardar las causas de error que presenten los datos y enviar el trámite a captación de datos.	<ol style="list-style-type: none"> <li>1. Si se detecta algún error en los datos                             <ol style="list-style-type: none"> <li>b) Omisión de caracteres</li> <li>c) Adición de caracteres</li> <li>d) Intercambio de datos</li> <li>e) Intercambio de caracteres en un dato</li> <li>f) Omisión de datos</li> </ol> </li> <li>1.1 Guardar la(s) causa(s) de error.</li> <li>1.2 Enviar trámite a la cola de captación de datos.</li> </ol>

**Tabla 17** Diseño de Caso de Prueba para el requisito Supervisar datos

#### 4.5.2 Pruebas Unitarias

Las pruebas unitarias se utilizan para ejecutar otro código fuente llamando directamente a los métodos de una clase, pasando los parámetros apropiados. Los métodos de pruebas unitarias residen en clases Test, que se almacenan en archivos de código fuente.



El objetivo de las pruebas unitarias es el aislamiento de partes del código y la demostración de que estas partes no contienen errores. (29)

Las pruebas unitarias realizadas a los servicios del sistema para validar que las salidas son las correctas y aseguran al desarrollador que su solución no presenta errores en la lógica de programación y que las respuestas son las correctas ante una entrada de datos determinada por el probador. A continuación (Ver **Fig. 18 Prueba Unitaria del Servicio Buscar Proceso**) se muestran las pruebas unitarias aplicadas a los servicios Buscar Procesos y Buscar Persona.

```
public void GetProccesRangeTimeTest ()
{
    DateTime StartDate = new DateTime(1999,12,31);
    DateTime EndDate = new DateTime(2012,12,31);
    string IdPerson = "1b53d480-28ae-6f18-e040-a8c0f7004d10";

    int expected = 2;
    int actual;
    actual =
SuinBusinessFactory.Instance.SearchProcessService.GetProccesbyRangeTime(StartDate,
EndDate,IdPerson).Count;
    Assert.AreEqual(expected, actual);
}
```

```
public void GetProccesByParamsTest ()
{
    Dictionary<Object, Object> parameters = new Dictionary<Object,
Object>();
    parameters.Add("IDPERSONA", "1b53d480-28ae-6f18-e040-a8c0f7004d10");

    int expected = 2;
    int actual;
    actual =
SuinBusinessFactory.Instance.SearchProcessService.ByParams(parameters).Count;
    Assert.AreEqual(expected, actual);
}
```

**Fig. 18 Prueba Unitaria del Servicio Buscar Proceso**

### 4.5.3 Pruebas de sistema

Las pruebas de sistema (pruebas de caja negra) se refiere a aquellas que se llevan a cabo sobre la interfaz del software. O sea, los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene. (30)

**4.5.3.1 Iteraciones de prueba**

Para comprobar la calidad del software se realizaron tres iteraciones de prueba sobre una aplicación funcional donde se validó la funcionalidad del sistema de acuerdo a los requisitos definidos. El objetivo fundamental de estas iteraciones es encontrar no conformidades y registrarlas para posteriormente resolverlas.

ID del Escenario	Tipos de Clases	Clases	Resultado Esperado	Resultado de la Prueba	Observaciones
1.1: Permitir la supervisión de datos.	Clases Válidas	Los campos a supervisar deben estar no editables.  Debe permitir la selección de los posibles errores que pueda tener un dato.	Permite supervisar los datos de un trámite.	Permite la supervisión de los datos.	
3.3. a Enviar trámite a captación de datos.		Debe mostrar la opción "Enviar a captación de datos" activada	Permite guardar las causas de error que presenten los datos y enviar el trámite a captación de datos.	Guarda las causas de error que presentan los datos y permite la corrección de los que son editables por el funcionario.	
1.1: Permitir la supervisión de datos.	Clases Inválidas	Los campos a supervisar se encuentran editables.  No permite la selección de los posibles errores que pueda tener un dato	No permite supervisar los datos de un trámite.		
3.3. a Enviar trámite a captación de datos.		Muestra la opción "Enviar a captación de datos" desactivada	No permite guardar las causas de error que presenten los datos y por tanto no envía el		

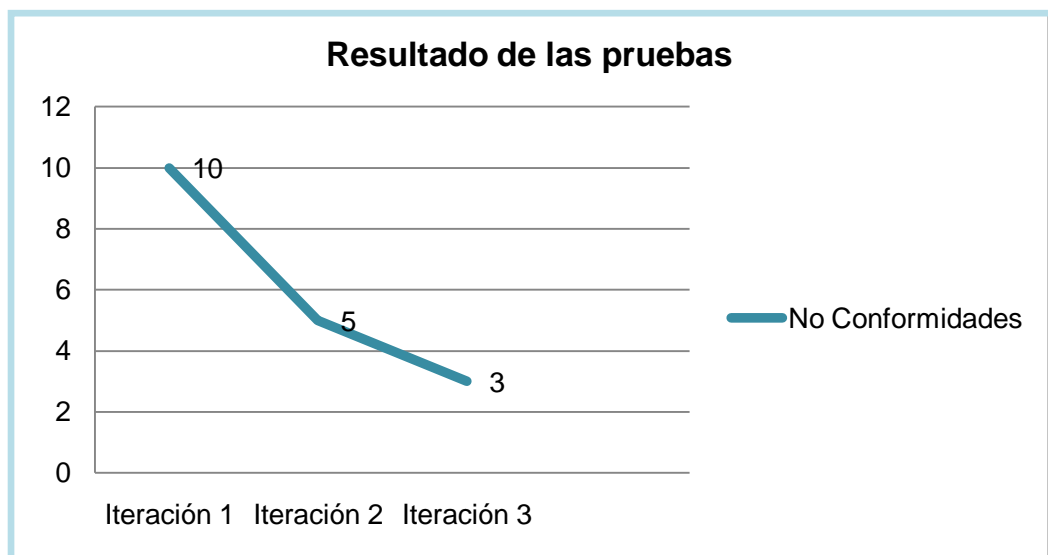
			trámite a captación de datos.		
--	--	--	-------------------------------------	--	--

**Tabla 18** Resultado de la tercera iteración del requisito Supervisar datos

#### 4.5.4 Resultado de las pruebas

Las pruebas aplicadas a la versión funcional del sistema arrojaron una serie de no conformidades que han sido representadas en la **Fig. 19 Resultado de las pruebas del sistema** y donde se observa cómo se les dio solución en cada iteración hasta quedar el sistema funcionando con todas sus capacidades.

Las pruebas unitarias aplicadas a los servicios resultaron satisfactorias para todos los parámetros que se les indicaron. Ver **Fig. 20 Resultado de la Prueba Unitaria realizada al Servicio Buscar Proceso**.



**Fig. 19** Resultado de las pruebas del sistema

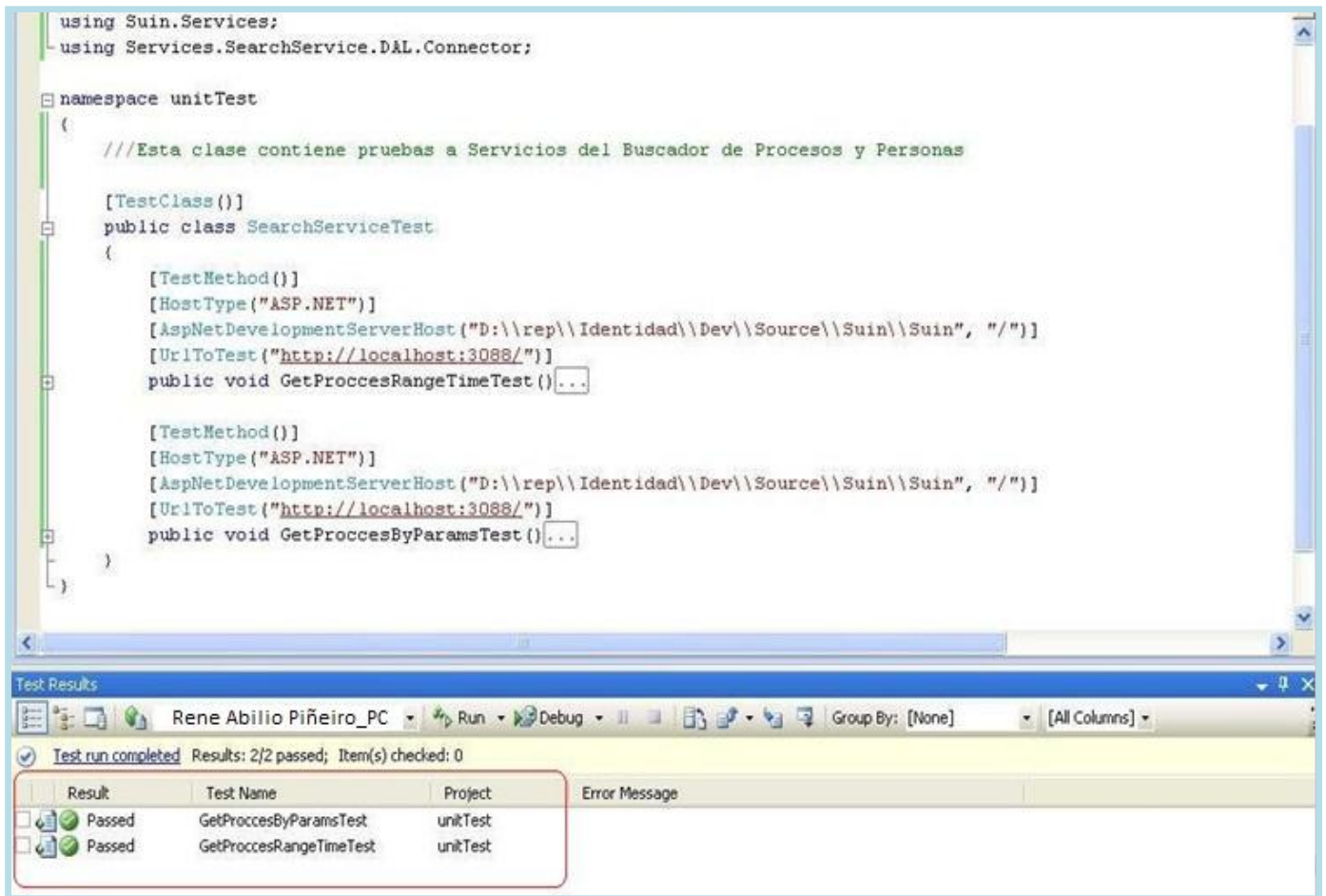


Fig. 20 Resultado de la Prueba Unitaria realizada al Servicio Buscar Proceso

#### 4.6 Conclusiones

- ❖ En el presente capítulo se llega a la solución final del software especificándose los componentes que lo conforman y que intervienen en la solución así como las relaciones físicas de los nodos que integran el Módulo de Supervisión y Control.
- ❖ Se abordan los temas referentes a los estándares de codificación y tratamiento de errores por los que se guió el equipo del proyecto para el desarrollo de la aplicación.
- ❖ Se evalúa el sistema de acuerdo a los diseños de casos de prueba definidos y que indicarán la calidad del mismo de acuerdo a las especificaciones del cliente.
- ❖ Se muestran las interfaces finales con las que interactúa el usuario de la aplicación luego de haber concluido el proceso de desarrollo del software.

### Conclusiones generales

Las empresas que en la actualidad utilizan sistemas informáticos para gestionar sus actividades garantizan claridad y rapidez en el manejo de las informaciones que gestionan. Con la realización del presente trabajo de diploma se ha desarrollado una primera versión del Módulo de Supervisión y Control del Sistema Único de Identificación Nacional, sistema que agrupa funcionalidades en un único software.

- ❖ Se hizo un extenso estudio de las aplicaciones que se dedicaban a la supervisión de trámites obteniéndose como resultado las bases para el desarrollo del Módulo de Supervisión y Control del SUIN.
- ❖ Se fundamentaron las tecnologías utilizadas para el desarrollo de la aplicación sentando las bases en las ventajas que ofrecían al desarrollo integrado y de acuerdo a peticiones del cliente.
- ❖ Se obtuvo el modelo de negocio de los procesos de tramitación en las oficinas CIRP relacionados a la supervisión de los trámites.
- ❖ Se expusieron los diagramas de proceso mejorado para mejor comprensión de las tareas a cumplir por parte de los desarrolladores, describiendo con estos diagramas los caminos a seguir.
- ❖ Se obtuvo el catálogo de requisitos funcionales del módulo y, para mejor comprensión, la descripción de ellos permitiendo así el análisis profundo de las funcionalidades que los requisitos deben cumplimentar y que fueron definidas con el cliente. Así como también se definieron los requisitos no funcionales que definen comportamientos específicos del sistema.
- ❖ Se definió la estructura organizacional de la arquitectura de la aplicación así como los módulos que lo conforman y, que unidos, integran el sistema con sus capacidades. Con esto se definieron clases y servicios que sustentan el desarrollo del software.
- ❖ Se probaron las funcionalidades del sistema a través del diseño de casos de prueba de la aplicación y las pruebas unitarias, permitiendo con esto validar el funcionamiento de los requisitos que fueron definidos.
- ❖ Se obtuvo una primera versión del Módulo de Supervisión y Control que garantiza un trámite seguro basándose en la revisión de los datos y la comprobación de la identidad de la persona mediante la impresión del Documento de Identidad Provisional.

### Recomendaciones

Luego de haber desarrollado una primera versión del Módulo de Supervisión y Control se recomienda:

- ❖ Hacer uso de la propuesta implementada que, integrada al SUIN, colaborará en la seguridad de la emisión de los documentos de identidad.
- ❖ Continuar el desarrollo de la aplicación dándole cumplimiento a los requisitos que no se han enmarcado en esta primera iteración.
- ❖ Definir nuevos elementos para la seguridad en la impresión del DIP de forma tal que se pueda obtener un documento cada vez más seguro, dificultando la posibilidad de falsificación y evitando el uso de elementos poco confiables.
- ❖ Integrar dentro del módulo de Supervisión un nuevo nivel de validación de la calidad de todos los parámetros de imágenes utilizando el componente que provee la herramienta Morpheus para garantizar el cumplimiento de estándares y recomendaciones internacionales.

### Referencias Bibliográficas

1. Jenesano. *Jenesano*. [En línea] Alcaldía Jenesano. [Citado el: 25 de Febrero de 2010.] <http://www.jenesano-boyaca.gov.co/glosario.shtml?apc=m-f1--&s=b>.
2. **Identidad-Venezuela, Equipo de Desarrollo Proyecto.** *Manuales de Usuario de Irregularidades*. Ciudad Habana : Universidad de las Ciencias Informáticas.
3. DATYS Ingeniería y Sistemas. *DATYS Ingeniería y Sistemas*. [En línea] Datys, 2009. [Citado el: 25 de Febrero de 2010.] <http://www.datys.cu/wpinfo/producto.aspx?12>.
4. **Muñoz Velazques, Caterine.** *Análisis y Diseño del Subsistema Matrícula para el Sistema de Gestión Académica Akademos 2.0*. Universidad de las Ciencias Informáticas. Ciudad Habana : Universidad de las Ciencias Informáticas, 2008. Tesis de grado.
5. **Morales, Ing. Pablo.** *Arquitectura de Procesos Para modelos de Workflow*. Montevideo - Uruguay : Comunidad de Workflow & BPM, Noviembre, 2002.
6. [En línea] Microsoft Corporation, 2005. [Citado el: 21 de Noviembre de 2009.] <http://guides.brucejmack.biz/MSF%20for%20CMMI%20Process%20Improvement/Process%20Guidance/Supporting%20Files/ProcessGuidance.htm>.
7. **Pérez, Juan Diego.** *Notaciones y lenguajes de procesos. Una visión global*. Departamento de Lenguajes y Sistemas Informáticos, Universidad de Sevilla. Sevilla. España : Universidad de Sevilla, 2009. pág. 100, Tesis de Doctorado.
8. DanySoft. Haciendo visible lo invisible. *DanySoft*. [En línea] 2010. [Citado el: 2 de Febrero de 2010.] <http://shop.danysoft.com/Altova-UModel>.
9. Monografías.com. *Monografías.com*. [En línea] 10 de Febrero de 2008. [Citado el: 21 de Noviembre de 2009.] <http://www.monografias.com/trabajos56/sistemas-bases-de-datos/sistemas-bases-de-datos.shtml>.
10. Oracle. *Oracle*. [En línea] 2010. [Citado el: 2 de Febrero de 2010.] <http://www.oracle.com/technology/products/database/oracle11g/index.html>.
11. **Katrib, Miguel, y otros.** *VisualStudio.NET 2008 desafía todo los retos*. [ed.] Octavio Hernández. 335. Ciudad Habana : Capitán San Luis.
12. DanySoft. *DanySoft*. [En línea] 25 de Febrero de 2008. [Citado el: 10 de Febrero de 2010.] <http://www.danysoft.com/bol/visualstudio2008.htm>.
13. Microsoft ASP.NET. *Microsoft ASP.NET*. [En línea] Microsoft Corporation, 2010. [Citado el: 15 de Febrero de 2010.] <http://www.asp.net/es>.

14. Canal Visual Basic .NET. *Canal Visual Basic .NET*. [En línea] canalvisualbasic.net, 2010. [Citado el: 15 de Febrero de 2010.] <http://www.canalvisualbasic.net/manual-net/c-sharp/#cSharp>.
15. Mossca.rog. *Mossca.rog*. [En línea] Buffa Sistemas. [Citado el: 11 de Febrero de 2010.] <http://www.mossca.org/>.
16. Globedia. *Globedia*. [En línea] HispaVista, 2010. [Citado el: 3 de Marzo de 2010.] <http://cu.globedia.com/windows-workflow-foundation>.
17. Microsoft.com. *Microsoft.com*. [En línea] Microsoft Corporation, 2010. [Citado el: 25 de Febrero de 2010.] [http://msdn.microsoft.com/en-us/library/aa697427\(VS.80\).aspx](http://msdn.microsoft.com/en-us/library/aa697427(VS.80).aspx).
18. MSDN. *MSDN*. [En línea] Microsoft Corporation, 2010. [Citado el: 25 de Febrero de 2010.] [http://msdn.microsoft.com/es-es/library/bb397926\(VS.90\).aspx](http://msdn.microsoft.com/es-es/library/bb397926(VS.90).aspx).
19. DanySoft. Haciendo visible lo invisible. *DanySoft*. [En línea] DanySoft, 2010. [Citado el: 10 de Marzo de 2010.] <http://www.codegear-shop.com/epages/62042259.sf?ObjectPath=/Shops/62042259/Products/%22Embarcadero%20ER/Studio%22>.
20. DanySoft. Haciendo visible lo invisible. *DanySoft*. [En línea] DanySoft, 2010. [Citado el: 7 de Marzo de 2010.] <http://shop.danysoft.com/Crystal-Reports-2008>.
21. Crystal Solution. *Crystal Solution*. [En línea] 2010. [Citado el: 22 de Marzo de 2010.] <http://www.crystalsolutions.com.ar/productos/crystalreports.html>.
22. *IEEE Standard Glossary of Software Engineering Terminology*. Institute of Electrical and Electronics Engineers. New York : IEEE Standards Board, 1900.
23. Application LifeCycle Solutions. *Application LifeCycle Solutions*. [En línea] Application LifeCycle Solutions, 2002-2007. [Citado el: 25 de Marzo de 2010.] <http://www.als-es.com/home.php?location=recursos/articulos/mejora-procesos-cmmi>.
24. **Standard, OASIS**. *Reference Model for Service Oriented Architecture 1.0*. s.l. : Adobe Systems Incorporated;MITRE Corporation;Fujitsu Laboratories of America Limited;Booz Allen Hamilton, 2006.
25. Definición.de. *Definición.de*. [En línea] Definición.de, 2008. [Citado el: 17 de Marzo de 2010.] <http://definicion.de/modelo-de-datos/>.
26. [En línea] 2010. [Citado el: 14 de Abril de 2010.] <http://www.dsi.uclm.es/asignaturas/42530/pdf/M2tema12.pdf>.
27. **Marca Huallpara, Hugo Michael y Quisbert Limachi, Nancy Susana**. *Trabajo de Investigación y Exposición*. 2010.



28. **Grupo de Aseguramiento de la Calidad Proyecto Identificación, Migración y Extranjería.** *Manual de Pruebas utilizando Microsoft Visual Studio Team System.* Universidad de las Ciencias Informáticas. Ciudad Habana : s.n., 2010.

29. Nova. *Nova.* [En línea] ACC System, 2010. [Citado el: 25 de Abril de 2010.]  
<http://www.novanebula.net/blog/archives/99-Unit-testing-pruebas-unitarias.html>.

30. SoftwareQATest.com. *SoftwareQATest.com.* [En línea] 10 de Enero de 2010. [Citado el: 25 de Abril de 2010.] <http://www.softwareqatest.com>.

31. Microsoft Corporation. *Microsoft Corporation.* [En línea] 2010. [Citado el: 11 de Febrero de 2010.]  
<http://www.microsoft.com/downloads/details.aspx?familyid=333325FD-AE52-4E35-B531-508D977D32A6&displaylang=es>.

### Glosario de términos

**ADO.NET:** es un conjunto de componentes del software que pueden ser usados por los programadores para acceder a datos y a servicios de datos. Es una parte de la biblioteca de clases base que están incluidas en el Microsoft .NET Framework., 39

**AFIS** (Automata Fingerprint Identification System): Sistema Automático de Identificación de Huellas Dactilares. Sistema informático compuesto de Hardware y Software integrados que permite la captura, consulta y comparación automática de huellas dactilares agrupadas por fichas decadactilares, o en forma de rastro latente (parte degradada de huella levantada en la escena de un crimen)., 27

**BPMN** (Business Process Management Notation): *Notación para el Modelado de Procesos de Negocio*. Permite el modelado de asuntos de negocio donde se presentan gráficamente las diferentes etapas del proceso del mismo. Esta notación coordina la secuencia de eventos que fluyen entre los diferentes procesos y participantes., 31

**C#:** es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, 32

**CLR** (Common Language Runtime): es el componente de máquina virtual de la plataforma .Net de Microsoft. Es la implementación del estándar Common Language Infrastructure (CLI) que define un ambiente de ejecución para los códigos de los programas., 36

**CMMI** (Capability Maturity Model Integration): Modelo de Madurez de las Capacidades, es un modelo para la mejora y evaluación de procesos para el desarrollo, mantenimiento y operación de sistemas de software., 30

**Eclipse:** es un entorno de desarrollo integrado de código abierto multiplataforma para desarrollar proyectos., 33

**Framework:** un framework, en el desarrollo de software es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.,35

**GRASP:** son patrones generales de software para asignación de responsabilidades, es el acrónimo de "General Responsibility Assignment Software Patterns" . Aunque se considera que más que patrones propiamente dichos, son una serie de "buenas prácticas" de aplicación recomendable en el diseño de software., 72

**IEEE** (Instituto de Ingenieros Electricistas y Electrónicos): es una asociación técnico-profesional mundial dedicada a la estandarización, entre otras cosas. Es la mayor asociación internacional sin fines de lucro formada por profesionales de las nuevas tecnologías, como ingenieros electricistas, ingenieros en electrónica, científicos de la computación, ingenieros en informática, ingenieros en biomédica, ingenieros en telecomunicación e Ingenieros en Mecatrónica., 49

**Java:** lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 90. El lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria., 32

**LINQ:** es un proyecto de Microsoft que agrega consultas nativas semejantes a las de SQL a los lenguajes de la plataforma .NET, inicialmente a los lenguajes Visual Basic .NET y C#., 39

**MSF** (Microsoft Solution Framework): Framework que brinda Microsoft para el guiar el desarrollo de sistemas. Es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos., 30

**.NET:** la plataforma .NET de Microsoft es un componente de software que puede ser añadido al sistema operativo Windows. Provee un extenso conjunto de soluciones predefinidas para necesidades generales de la programación de aplicaciones, y administra la ejecución de los programas escritos específicamente con la plataforma. Esta solución es el producto principal en la oferta de Microsoft, y pretende ser utilizada por la mayoría de las aplicaciones creadas para la plataforma Windows., 34

**Popfly:** es un sitio web que permite al usuario crear páginas web, snippets, y Mashups(aplicación web híbrida) usando Runtimes para Aplicaciones ricas de internet basadas en Microsoft Silverlight., 36

**Servicios Web:** Colección de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores., 38

**Silverlight:** es una estructura para aplicaciones web que agrega nuevas funciones multimedia como la reproducción de vídeos, gráficos vectoriales, animaciones e interactividad, en forma similar a lo que hace Adobe Flash, 36

**SQL** (Structured Query Language): es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre las mismas., 41

**UML** (Unified Modeling Language): Lenguaje Unificado de Modelado es un lenguaje de modelado de sistemas de software., 32

**VB .NET:** Visual Basic .NET es un lenguaje de programación orientado a objetos que se puede considerar una evolución de Visual Basic implementada sobre el framework .NET., 32

**Windows CE:** Sistema operativo de Microsoft incrustado modular de tiempo real para dispositivos móviles de 32-bits inteligentes y conectados., 35

**Workflow:** sistemas que, de manera completa, definen, gestionan, controlan y ejecutan flujos de trabajo en el contexto de procesos de negocio, a través de la ejecución de software, cuyo orden de ejecución es controlado por una representación computarizada del proceso de negocio., 30

**Workstream:** secuencia de actividades de trabajo, 31

**XML** (Extensible Markup Language): Metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C). Es una manera de definir lenguajes para diferentes necesidades. XML es una tecnología sencilla que tiene a su alrededor otras que la complementan y la hacen mucho más grande y con unas posibilidades mucho mayores. Tiene un papel muy importante en la actualidad ya que permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil., 32