

**UNIVERSIDAD MAYOR DE SAN ANDRÉS**  
FACULTAD DE CIENCIAS PURAS Y NATURALES  
CARRERA DE INFORMATICA



**PROYECTO DE GRADO**  
“SISTEMA BIOMÉTRICO DE CONTROL  
DE ASISTENCIA Y PLANILLAS DE PAGO”

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMÁTICA  
MENCIÓN INGENIERÍA DE SISTEMAS INFORMÁTICOS

**POSTULANTE: EGBERTO TOLA FLORES**

Tutor: Lic. Mario Loayza Molina M. Sc.  
Revisor: Lic. Hugo Javier Reyes Pacheco

LA PAZ BOLIVIA  
2007

## **Dedicatoria**

Este trabajo se lo dedico de  
manera especial a mis padres  
y hermanos, que me apoyaron  
en cada momento.

## **Agradecimientos**

A Dios, por su bendición.

Mis mas sinceros agradecimientos al lic. Mario Loayza Molina por su valioso asesoramiento y apoyo para la culminación del mi proyecto de grado.

De igual manera agradezco a mi revisor de proyecto Lic. Javier Reyes Pacheco, por sus consejos y orientación.

A todos los docentes de mi carrera, por compartir sus conocimiento.

A mi papá Juan Cleto Tola y a mi mamá Juliana Flores, por su apoyo y comprensión.

A mis hermanos, por el apoyo que me brindaron.

## RESUMEN

La necesidad de tener una forma de identificar de manera única al personal de cualquier institución, exige a implementan una serie de métodos para optimizar el control del personal. Hoy en día, el más avanzado es el método biométrico que utiliza dispositivos electrónicos que captura patrones que identifica de manera única a las personas. Viendo esta necesidad llevo acabo este proyecto utilizando el método biométrico de huella dactilar, este método utiliza un dispositivo electrónico que captura la huella dactilar y con ello genera un código de identificación.

En este proyecto se describe el Proceso Unificado de Rational (RUP), el cual trabaja con un modelo que disminuye los riesgos en la implementación del software. El RUP utiliza el Lenguaje Unificado de Modelado UML para preparar todos los esquemas del sistema en la etapa de diseño.

El Gobierno Municipal de El Alto tiene disponible para su uso, el leguaje Visual Basic .NET, por esta razón y por ser flexible en el uso de puertos USB, el sistema propuesto se programara en este lenguaje.

## ÍNDICE

|                     |     |
|---------------------|-----|
| Dedicatoria.....    | i   |
| Agradecimiento..... | ii  |
| Resumen.....        | iii |

### **CAPITULO I**

#### **ANTECEDENTES Y MARCO REFERENCIAL**

|                                     |   |
|-------------------------------------|---|
| 1. 1 INTRODUCCIÓN.....              | 1 |
| 1.2 ANTECEDENTES.....               | 1 |
| 1.3 PLANTEAMIENTO DEL PROBLEMA..... | 2 |
| 1.3.1 Descripción del problema..... | 2 |
| 1.3.2 Definición del problema.....  | 3 |
| 1.4 OBJETIVOS.....                  | 3 |
| 1.4.1 Objetivo general.....         | 3 |
| 1.4.2 Objetivos específicos.....    | 3 |
| 1.5 LIMITE Y ALCANCES.....          | 4 |
| 1.5.1 Alcances.....                 | 4 |
| 1.5.2 Limites.....                  | 4 |
| 1.6 JUSTIFICACIÓN.....              | 5 |
| 1.6.1 Técnica.....                  | 5 |
| 1.6.2 Social.....                   | 5 |
| 1.6.3 Económica.....                | 5 |

### **CAPITULO II**

#### **MARCO TEÓRICO**

|  |   |
|--|---|
| IDENTIFICACIÓN BIOMÉTRICA.....               | 7 |
| Sistemas de Autenticación Biométrica.....    | 7 |
| Cómo funcionan los sistemas biométricos..... | 8 |
| Técnicas Biométricos.....                    | 9 |

|   |           |
|---|-----------|
| Historia de la huella digital.....                                      | 11        |
| Fase del proceso biométrico.....  | 13        |
| Inscripción .....   | 13        |
| Identificación .....  | 14        |
| 2.1.6 Técnicas de obtención de la huella digital...14                   |           |
| 2.1.7 Puntos de fortaleza y debilidad.....                              | 16        |
| <b>2.2 METODOLOGÍA APLICADA .....</b>                                   | <b>17</b> |
| 2.2.1 Introducción.....   | 17        |
| 2.2.2 Metodología del proceso unificado del desarrollo de software..... | 17        |
| 2.2.2.1 Ciclo de vida del Proceso Unificado.....                        | 19        |
| 2.3 UML.....  | 20        |
| 2.3.1 Elementos.....  | 21        |
| 2.3.2 Relaciones.....   | 25        |
| 2.3.3 Diagramas.....  | 26        |
| 2.4 CALIDAD DEL SOFTWARE.....   | 28        |
| 2.4.1 Características de calidad.....                                   | 28        |
| <b>CAPÍTULO III</b>   |           |
| <b>DESARROLLO DEL PROYECTO</b>  |           |
| 3.1 ANÁLISIS DE REQUERIMIENTO.....                                      | 32        |
| 3.2 ENCONTRAR ACTORES Y CASOS DE USO.....                               | 33        |
| 3.3.1 Encontrar actores.....  | 33        |
| 3.3.2 Encontrar casos de uso.....                                       | 33        |
| 3.3.3 Priorizar casos de uso.....                                       | 34        |
| 3.3 MODELO DEL NEGOCIO.....   | 37        |
| 3.4 CASOS DE USO.....   | 37        |
| 3.4.1 Detallar casos de uso.....  | 38        |
| 3.5 ANÁLISIS Y DISEÑO.....  | 45        |
| 3.4.1 Modelo de datos.....  | 45        |
| 3.6 IMPLEMENTACIÓN.....   | 46        |
| 3.5.1 Diagrama global de paquetes.....                                  | 46        |
| 3.5.2 Diagrama de componentes.....                                      | 46        |

|   |    |
|---|----|
| 3.5.3 Diagrama de despliegue.....         | 48 |
| 3.5.4 Interfaces graficas de usuario..... | 48 |
| 3.7 FACTORES DE CALIDAD ISO 9126.....     | 53 |
| 3.7.1 Confiabilidad.....                  | 53 |
| 3.7.2 Funcionalidad.....                  | 55 |
| 3.7.3 Facilidad de uso.....               | 58 |
| 3.7.4 Facilidad de mantenimiento.....     | 59 |
| 3.7.5 Portabilidad.....                   | 59 |

## **CAPÍTULO IV**

### **CONCLUSIONES**

|                       |    |
|-----------------------|----|
| 4.1 CONCLUSIONES..... | 61 |
|-----------------------|----|

### **REFERENCIAS BIBLIOGRÁFICAS**

### **ANEXOS**

### **DOCUMENTACIÓN**



## ÍNDICE DE FIGURAS

|  |    |
|--|----|
| <i>Figura 2.1. Técnicas biométricas actuales</i> .....             | 10 |
| Figura 2.2: Fases de proceso biométrico.....                       | 13 |
| Figura 2.3: Huella dactilar con sus minucias extraídas.....        | 14 |
| Figura 2.4: Técnicas de obtención de la huella digital.....        | 18 |
| Figura 2.5. Muestra la impresión digital del mismo dedo.....       | 16 |
| Figura 2.6: Calidad de impresiones digitales.....                  | 27 |
| Figura 2.7: Ciclo de vida del Proceso Unificado.[Booch, 2000]..... | 19 |
| Figura 2.8: Historia de UML.....                                   | 20 |
| Figura 3.1 Subsistemas.....  | 34 |
| Figura 3.2: Modelo de Casos de Uso del Negocio.....                | 35 |
| Figura 3.3: Modelo de objetos de Marcar Asistencia.....            | 35 |
| Figura 3.4: Modelo de objetos de Controlar Asistencia.....         | 36 |
| Figura 3.6: Diagrama Casos de Uso.....                             | 37 |
| Figura 3.7: Caso de uso Marcar Asistencia.....                     | 38 |
| Figura 3.8: Caso de uso Registrar Empleado.....                    | 41 |
| Figura 3.9: Caso de uso generar reportes.....                      | 42 |
| Figura 3.10: Caso de uso Planificar Ausencia.....                  | 43 |
| Figura 3.11: Caso de uso Actualizaciones.....                      | 44 |
| Figura 3.12: Modelo de datos.....                                  | 45 |
| Figura 3.14: Diagrama global de paquetes.....                      | 46 |
| Figura 3.15: Diagrama de componentes comunes.....                  | 46 |
| Figura 3.16: Diagrama de componentes control de personal.....      | 47 |
| Figura 3.17: Diagrama de componentes planilla de pagos.....        | 47 |
| Figura 3.18: Diagrama de despliegue.....                           | 48 |



**CAPITULO I  
ANTECEDENTES  
Y MARCO REFERENCIAL**

## **CAPITULO II**

### **ANTECEDENTES Y MARCO REFERENCIAL**

#### **1.1 INTRODUCCIÓN**

En una institución el control de asistencia es muy importante, por que los salarios se pagan según la asistencia registrada, por ello se busca medios confiables para el control del personal.

El mecanismo de control de asistencia más común es el reloj mecánico marcador de tarjetas o las tarjetas magnéticas, pero estos no son confiables por el riesgo de sustitución de personas. Todos estos inconvenientes pueden ser desechados ya que el ser humano tiene características que lo hacen único, estas características son: huellas digitales, la voz, el iris del ojo que pueden ser aprovechados para la identificación de las personas.

Entre todas las técnicas, el más accesible es la técnica de reconocimiento basado en la huella digital.

#### **1.2 ANTECEDENTES**

La Dirección de Gestión de Capital Humano del Gobierno Municipal de El Alto, esta compuesto por varias áreas de las cuales dos son de interés para este proyecto:

- Control de Personal, que cumple la función de supervisar y efectuar el control de entradas, salidas, asistencia y permanencia en el puesto de trabajo del personal del municipio.
  
- Planillas que cumple la función de elaborar y procesar planillas conforme a los requerimientos de las leyes y normas establecidas, además de elaborar y procesar boletas de pago y medios magnéticos para su remisión al banco.

El Control del Personal es uno de los motores principales de esta unidad, debido a que las planillas se realizan de acuerdo a los datos registrados por el control del personal. En la actualidad existen algunos problemas en el marcado de asistencia y la generación de planillas de pago, por que no tienen un sistema informático que apoye a esta unidad del Municipio.

Existen en el mercado sistemas de control de asistencia y planillas, pero no responden a los requerimientos del Municipio, razón por lo que es necesario realizar un sistema que este a la medida del usuario.

### **1.3 PLANTEAMIENTO DEL PROBLEMA**

#### **1.3.1 Descripción del problema**

En la Unidad de Control de asistencia y planillas de pago, el control de asistencia se realiza actualmente mediante un marcador mecánico de tarjetas, los datos de las tarjetas son pasados manualmente a Excel para obtener las planillas de pago.

Los datos obtenidos por el marcador de tarjetas no son fiables, además ocasiona pérdida de tiempo en la transferencia de datos de asistencia para la generación de planillas.

La falta de un mecanismo eficiente de control de asistencia y obtención de planillas de pago, demoran en la obtención de informes y reportes de planillas de pago. Si no se dan pronta solución a estos problemas la asistencia del personal será irregular y existirá retrasos en la generación de planillas de pago, es por esto que se ve la necesidad de implementar un sistema que apoye al control de asistencia y la generación de planillas de pago.

### **1.3.2 Definición del problema**

En la Unidad de Control de asistencia y planillas de pago los reportes de los empleados son generados manualmente, el cual dificulta el monitoreo y control eficiente de estas actividades.

## **1.4 OBJETIVOS**

### **1.4.1 Objetivo general**

Diseñar e implementar un “Sistema Biométrico de Control de Asistencia y Planillas de Pago” para el Gobierno Municipal de El Alto.

### **1.4.2 Objetivos específicos**

Para resolver las necesidades de la Unidad de Control de Asistencia y Planillas de pago, se plantean los siguientes objetivos específicos:

- Diseñar una arquitectura sólida del sistema de acuerdo a los requerimientos y una base de datos apropiada.
- Brindar una herramienta que facilite el control de asistencia y además se tenga una disponibilidad inmediata de las planillas de pago.
- Reducir errores en la generación de planillas.
- Evitar suplantación de personas a la hora de marcar asistencias.
- Optimizar el tiempo de procesamiento de datos para obtener planillas de pago.
- Generar reportes, informes e historial de asistencias.

- Diseñar una interfaz de usuario maquina amigable.
- Implementar un modulo de información al personal del Municipio.

## **1.5 LIMITE Y ALCANCES**

### **1.5.1 Alcances**

El principal objetivo de este proyecto es desarrollar un sistema para facilitar el manejo de la información de las planillas de pago con la obtención de datos de asistencias del personal con un método fiable, para la Unidad de Control de asistencia y Planillas de Pago, se consideraran las siguientes funciones:

- Función de registro de huellas y asignación de usuarios.
- Genera reportes del movimiento de los usuarios como: cambios de turnos, inasistencias, retardos, etc.
- Liquidación Mensual
- Reporte de Planillas de Pago
- Obtiene reportes de asistencia.
- Genera reportes de la hora de entrada y salida del personal de acuerdo al horario asignado.
- El módulo de planillas puede recuperar la información de asistencia y calcular los descuentos por faltas y tardanzas.

### **1.5.2 Limites**

El sistema no se realizo para toda la Dirección de Gestión de Capital Humano se restringe a lo que es control de asistencia y planillas de pago.

## **1.6 JUSTIFICACIÓN**

### **1.6.1 Técnica**

La unidad para la cual realizo el proyecto, cuenta con la licencia del lenguaje VB.NET y equipos con las características necesarios para la implantación del sistema. Además de que el lenguaje presenta una forma fácil y rápida de programar, el sistema será desarrollado en el lenguaje VB.NET.

### **1.6.2 Social**

Con el presente proyecto la Unidad de Control de Personal y Planillas de Pago podrá brindar rápidamente informaciones solicitadas por los funcionarios, además como las planillas de pago se obtendrán en menor tiempo, será posible pagar sus sueldos en el periodo establecido.

### **1.6.3 Económico**

La ejecución del proyecto permitirá optimizar el control de personal y obtención de planillas de pago en forma automática. Logrando con esto reducir la intervención de recurso humano para realizar el control del personal y la obtención de planillas de pago.



## **CAPITULO III MARCO TEÓRICO**

## CAPITULO II MARCO TEÓRICO

### 2.1 IDENTIFICACIÓN BIOMÉTRICA

#### 2.1.1 Sistemas de autenticación biométrica

A pesar de la importancia de la criptología en cualquiera de los sistemas de identificación de usuarios vistos, existen otra clase de sistemas en los que no se aplica esta ciencia, o al menos su aplicación es secundaria. Es más, parece que en un futuro no muy lejano estos serán los sistemas que se van a imponer en la mayoría de situaciones en las que se haga necesario autenticar un usuario: son más amigables para el usuario (no va a necesitar recordar *passwords* o números de identificación complejos, y, como se suele decir, el usuario puede olvidar una tarjeta de identificación en casa, pero nunca se olvidará de su mano o su ojo) y son mucho más difíciles de falsificar que una simple contraseña o una tarjeta magnética.

Estos sistemas son los denominados **biométricos**, basados en características físicas del usuario a identificar. El reconocimiento de formas, la inteligencia artificial y el aprendizaje son las ramas de la informática que desempeñan el papel más importante en los sistemas de identificación biométricos; la criptología se limita aquí a un uso secundario, como el cifrado de una base de datos de patrones retinales, o la transmisión de una huella dactilar entre un dispositivo analizador y una base de datos.

Los dispositivos biométricos tienen tres partes principales; por un lado, disponen de un mecanismo automático que lee y captura una imagen digital o analógica de la característica a analizar. Además disponen de una entidad para manejar aspectos como la compresión, almacenamiento o comparación de los datos capturados con los guardados en una base de datos, y también ofrecen una interfaz para las aplicaciones que los utilizan. El proceso general de autenticación sigue unos pasos comunes a todos los modelos de autenticación

biométrica: **captura** o lectura de los datos que el usuario a validar presenta, **extracción** de ciertas características de la muestra (por ejemplo, las minucias de una huella dactilar), **comparación** de tales características con las guardadas en una base de datos, y **decisión** de si el usuario es válido o no. Es en esta decisión donde principalmente entran en juego las dos características básicas de la fiabilidad de todo sistema biométrico.

### 2.1.2 Cómo funcionan los sistemas biométricos

Los sistemas biométricos se componen de un hardware y un software; el primero captura la característica concreta del individuo y el segundo interpreta la información y determina su aceptabilidad o rechazo, todo en función de los datos que han sido almacenados por medio de un registro inicial de la característica biométrica que mida el dispositivo en cuestión. Ese registro inicial o toma de muestra es lo que determina la eficacia del sistema. En el caso de las huellas dactilares, un usuario coloca el dedo en un sensor que hace la lectura digital de su huella, después, el programa guardará la información como un modelo; la próxima vez que ese usuario intente acceder al sistema deberá repetir la operación y el software verificará que los datos corresponden con el modelo. El mismo principio rige para la identificación por el iris/retina, con ayuda de videocámara, el rostro, la mano completa, etc. Las tasas de exactitud en la verificación dependen en gran medida de dos factores: el cambio que se puede producir en las personas, debido a accidentes o a envejecimiento, y las condiciones ambientales, como humedad en el aire, suciedad y sudor, en especial en la lectura que implique el uso de las manos.

En cuanto a qué partes del cuerpo son las más adecuadas para su utilización en identificación biométrica, aunque en principio cualquiera sería susceptible de ser usada, para su elección se atiende a criterios prácticos concretos. Lo ideal es que se trate de una característica física robusta, es decir, no sujeta a grandes cambios; que sea lo más distintiva posible en relación con el resto de la

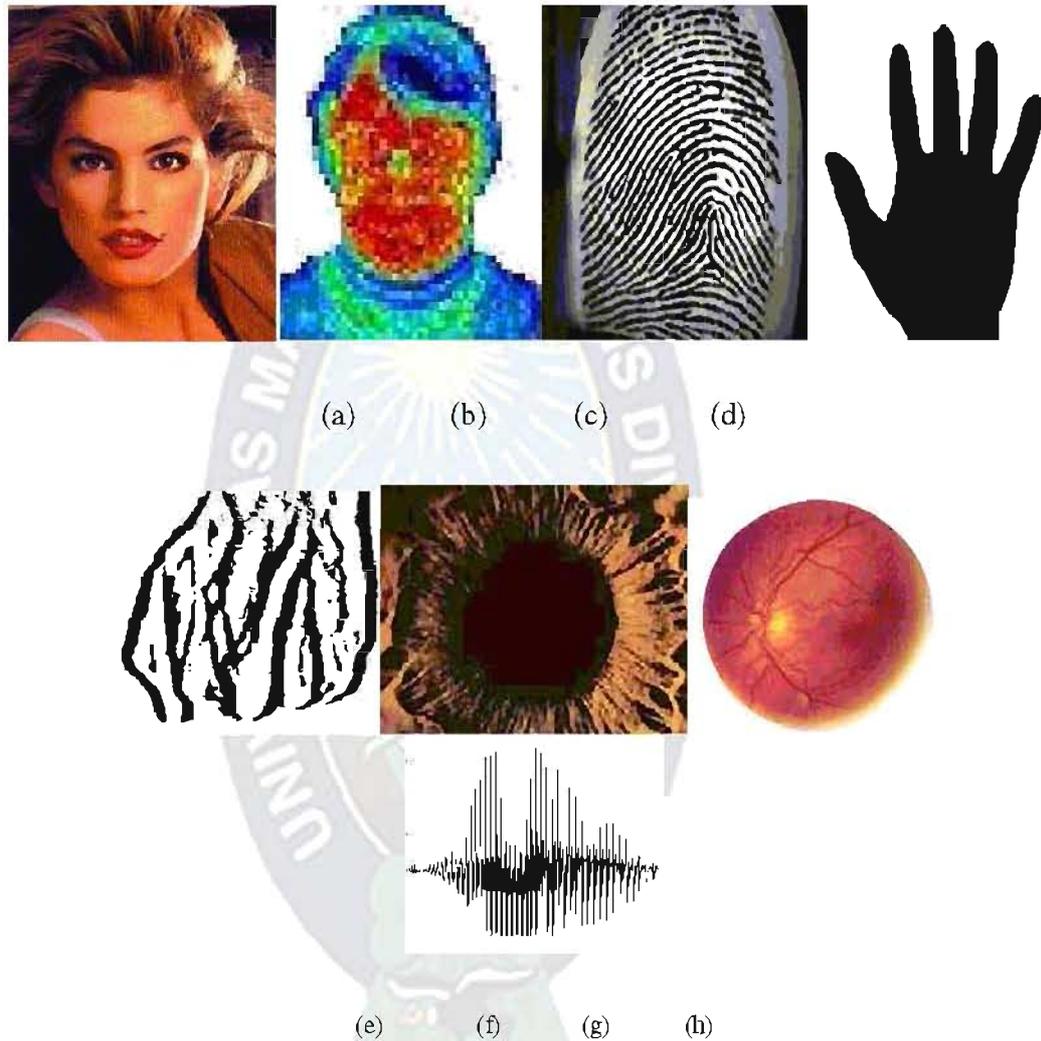
población, que sea una zona accesible, disponible y, por supuesto aceptable por el usuario.

Pro último, hay que hacer una distinción entre aquellos dispositivos que miden el comportamiento y los que miden una característica fisiológica. Entre los primeros se encuentran el análisis de la dinámica de la firma y el del golpe en el teclado; los segundos incluyen la huella dactilar, la geometría de la mano y el dedo, la termografía facial y la exploración del iris o la retina. El reconocimiento de la voz es un parámetro biométrico basado en ambos análisis, el fisiológico que determina la zona vocal y el de comportamiento del lenguaje y las palabras usadas. Evidentemente aquellos dispositivos que se basen en el comportamiento requieren de la cooperación del usuario, mientras que se puede identificar fisiológicamente a cualquiera sin su cooperación e incluso sin su conocimiento, como en el caso de la imagen captada por una videocámara.

### **2.1.3 Técnicas Biométricas**

En la actualidad existen sistemas biométricos que basan su acción en el reconocimiento de diversas características. Las técnicas biométricas más conocidas son:

1. Rostro
2. Termograma del rostro
3. Huellas dactilares
4. Geometría de la mano
5. Venas de las manos
6. Iris
7. Patrones de la retina
8. Voz



*Figura 2.1 . Técnicas biométricas actuales: (a) Rostro, (b) Termograma Facial, (c) Huella dactilar, (d) Geometría de la mano, (e) Venas de la mano, (f) Iris, (g) Patrones de la retina, (h) Voz e (i) Firma.*  
 Fuente: [http://www2.ing.puc.cl/~iing/ed429/sistemas\\_biometricos.htm](http://www2.ing.puc.cl/~iing/ed429/sistemas_biometricos.htm)

Los sistemas basados en reconocimiento de huellas son relativamente baratos (en comparación con otros biométricos, como los basados en patrones retíneales), por eso es esta técnica que se usara para el desarrollo del proyecto de grado.

#### 2.1.4 Historia de la huella digital

- Sir William Herschel – 1856. Fue el primer inglés que empezó a usar huellas digitales en Julio de 1858, cuando Sir William Herschel , principal magistrado del distrito en Jungipoor, India, uso huellas digitales para los contratos con los nativos. Solo como un capricho y sin pensar en la identificación personal, Herschel tenía a Rajyadhar Konai, hombre de negocios local, el cual puso una impresión de su mano en la parte trasera de un contrato. La idea era simplemente para asustarlo, en caso de que él decidiera negar su firma, se tenía una firma natural. Hersechel hizo un habito el requerir las impresiones de la palma, mas adelante solo requirió la impresión del índice derecho y los dedos medios, para cada contrato que hizo con la gente local. El hacer un contacto mas personal con el papel era para ellos la creencia de que era más poderosa y obligatoria que si solamente se firmara. Así es que el moderno sistema de huellas digitales se fundamenta no en bases científicas si no que las bases fueron de superstición. Conforme su colección de huellas digitales creció, Hersechel noto que las impresiones entintadas podrían probar o refutar de hecho una identidad. Mientras que su experiencia con huella digital era obviamente limitada, Sir Herschel tenía la convicción propia de que las huellas digitales eran únicas al individuo, así como permanentes a lo largo de la vida del individuo.
- Dr. Henry Faulds – 1880. Durante el 1870, el Dr. Henry Faulds, Cirujano-Superintendente del hospital de Tsukiji en Tokio, Japón, tomó el estudio de " piel surcos " después de notar marcas de dedos en especímenes de cerámica prehistórica. El Dr. Faulds no solamente reconoció la importancia de las huellas digitales como medios de identificación, sino que también como medio de clasificación. En 1880 Faulds remitió una

explicación de su sistema de clasificación y de una muestra de las formas que él había diseñado para las impresiones entintadas, a sir Charles Darwin. Darwin, en edad avanzada y mala salud, informó al Dr. Faulds que él no podría ayudarlo, pero prometido pasar los materiales a su primo, Francis Galton. También en 1880, el Dr. Faulds publicó un artículo en el diario científico, " Nature " (naturaleza). Él habla de las huellas digitales como medios de identificación personal, y el uso de la tinta de las impresoras como método para obtener tales huellas digitales. También lo acreditan con la primera identificación de una huella digital grasienta a la izquierda en una botella del alcohol.

- Gilbert Thompson – 1882. En 1882, Gilbert Thompson, utilizó sus propias huellas digitales en un documento para prevenir la falsificación. Éste es el primero uso conocido de huellas digitales en los Estados Unidos.
- El sistemático de huellas digitales se uso primero en los Estados Unidos por la Comisión de Servicio Civil de Nueva York. El Dr. Henry P. DeForrest inicia la huella dactilar.

## 2.1.5 Fase del proceso biométrico

La fase del proceso biométrico consta de inscripción y identificación.

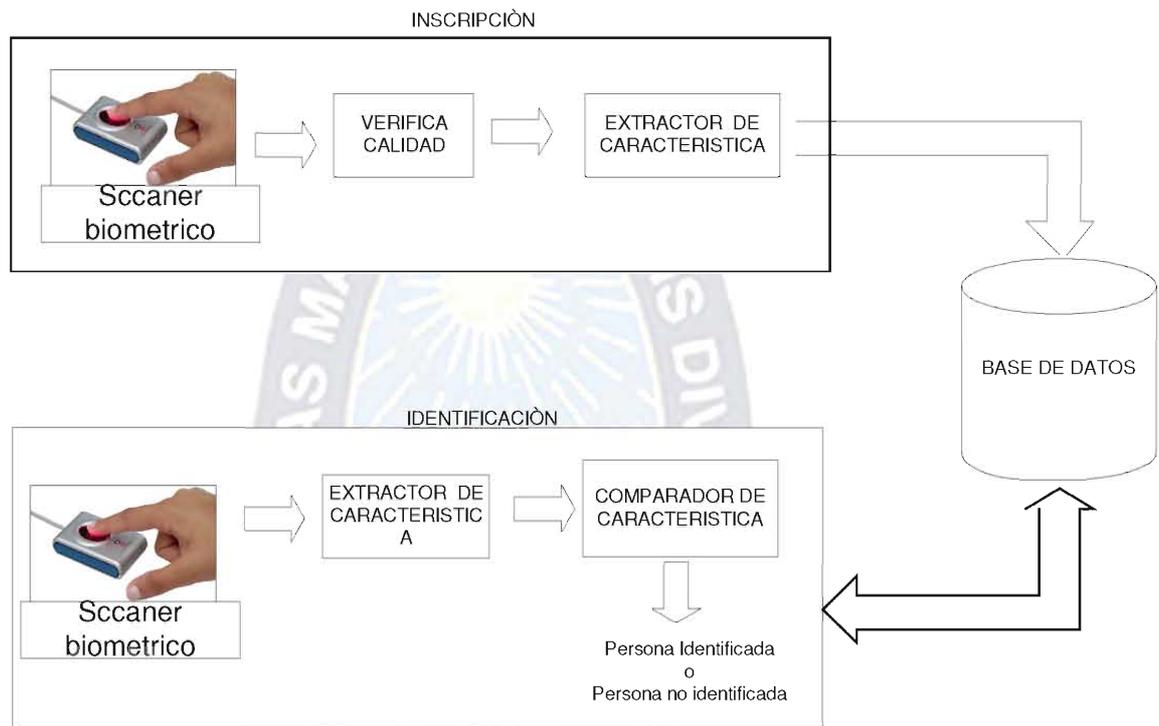


Figura 2.2: Fases de proceso biométrico Fuente: Modificación de Fuente: [http://www2.ing.puc.cl/~iing/ed429/sistemas\\_biometricos.htm](http://www2.ing.puc.cl/~iing/ed429/sistemas_biometricos.htm)

### 2.1.5.1 Inscripción

El módulo de inscripción es la primera fase del proceso biométrico, se encarga de adquirir y almacenar la información proveniente del indicador biométrico con el objeto de poder contrastar a ésta con la proporcionada en ingresos posteriores al sistema. Las labores ejecutadas por el módulo de inscripción son posibles gracias a la acción del lector biométrico y del extractor de características

El primero se encarga de adquirir datos relativos al indicador biométrico elegido y entregar una representación en formato digital de éste. El segundo extrae, a

partir de la salida del lector, características representativas del indicador. El conjunto de características anterior, que será almacenado en una base de datos central u otro medio y que será utilizada en las labores de identificación al ser comparada con la información proveniente del indicador biométrico en el punto de acceso.

#### **2.1.5.2 Identificación**

El módulo de identificación es el responsable del reconocimiento de individuos, por ejemplo en una aplicación de control de acceso. El proceso de comprobación comienza cuando el lector biométrico captura la característica del individuo a ser identificado y la convierte a formato digital, para que a continuación el extractor de características produzca una representación compacta con el mismo formato de los que almacenados en la fase de inscripción. La representación resultante de identificación es enviada al comparador de *características* que confronta a éste con uno o varias muestras extraídas en la fase de inscripción.

#### **2.1.6 Técnicas de obtención de la huella digital**



Figura 2.3: Huella dactilar con sus minucias extraídas.  
©1998 Idex AS, <http://www.idex.no/>.

Fuente: <http://es.tldp.org/Manuales-LuCAS/doc-unixsec/unixsec-html/node113.html>

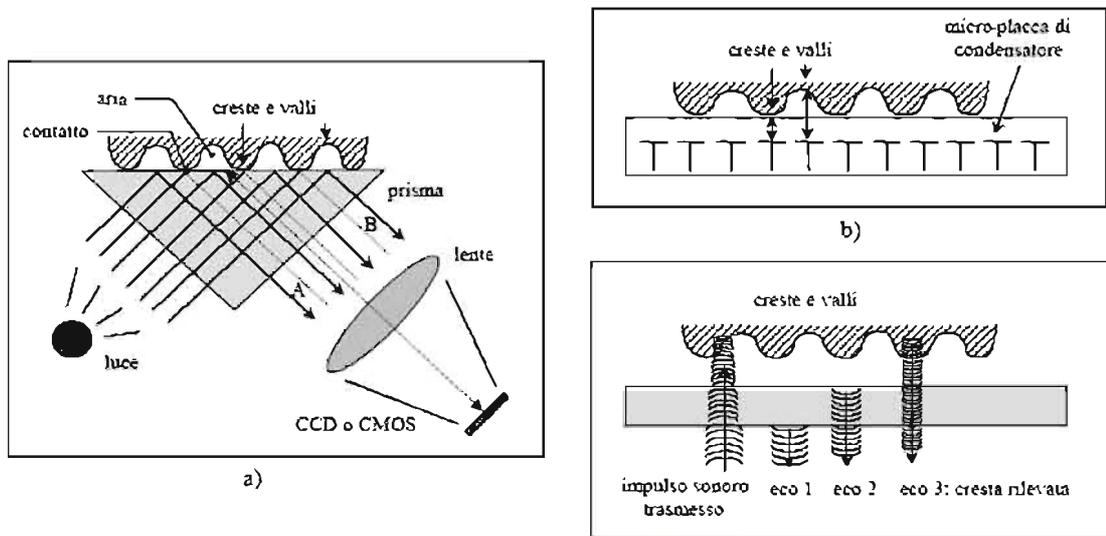


figura 2.4: a) por medio de un sensor óptico, b) por medio de un sensor capacitivo; c) por medio de un sensor ultrasonido  
 Fuente: <http://es.tldp.org/Manuales-LuCAS/doc-unixsec/unixsec-html/node113.html>

En la técnica del sensor óptico los datos se obtiene según los puntos de contacto del dedo con la superficie de un prisma, de acuerdo a esto se tendrá intensidades de luces diferentes en cada punto de la lente.

En la técnica del sensor capacitivo el micro capa de condensadores puede captar las distancias diferentes entre las crestas y valles y convertirla en las señales eléctricos.

En la técnica del sensor del ultrasonido que realiza una clase de ecografía al dedo, variado según la estructura del epidermis de un dedo.

La figura (2.5) muestra la impresión digital del mismo dedo, adquirido con los sistemas comerciales diferentes. Muestran claramente como los sensores diferentes pueden adquirir imágenes de calidad diferente y riqueza de detalles.



Figura 2.5. Muestra la impresión digital del mismo dedo

Fuente: <http://es.tldp.org/Manuales-LuCAS/doc-unixsec/unixsec-html/node113.html>

### 2.1.7 Puntos de fortaleza y debilidad

En tabla 2.1 se identifica algunos puntos de fortaleza y debilidad de los sistemas biométricos basados en las huellas digitales:

| Fortaleza  | Debilidad  |
|--|--|
| <ul style="list-style-type: none"> <li>• La tecnología consolidada</li> <li>• La exactitud elevada</li> <li>• Los dispositivos de adquisición de dimensiones pequeñas</li> <li>• Los costos reducidos</li> </ul> | <ul style="list-style-type: none"> <li>• Pueden existir la dificultad debido al espesor reducido de las crestas epidérmicas.</li> <li>• Datos erróneos por tener demasiada humedad o polvorientos los dedos.</li> <li>• La desconfianza de algunos asuntos para los factores psicológicos</li> </ul> |

Tabla 2.1: fortaleza y debilidad de los sistemas biométricos basados en las huellas digitales

Fuente: Elaboración propia

En figura 2.6 se muestran dos impresiones digitales de calidad diferente adquiridas por dos casos diferentes: la impresión del inciso b) es de calidad mala; las impresiones de calidad mala es más sujeto a los errores. Otro de los casos es que tenga los dedos muy húmedos.



Figura 2.6: Calidad de impresiones digitales a) normal; b) húmedos

Fuente: <http://www.tecnociencia.es/monograficos/biometria/biometria2.htm>

## 2.2 METODOLOGÍA APLICADA

### 2.2.1 Introducción

La realización del modelo se fundamenta en la metodología de ingeniería de software conocida como Proceso Unificado (o Proceso Unificado de Rational), la cual es una metodología orientada a objetos e intenta reunir las mejores características de otros métodos como el OMT (Object Modeling Technique, Técnica de Modelado de Objetos) y el OOSE (Object – Oriented Software Engineering, Ingeniería del Software Orientada a Objetos) entre otros.

El Lenguaje Unificado de Modelado (UML) se utiliza para presentar de una forma clara y precisa el seguimiento de la metodología.

### 2.2.2 Metodología del Proceso Unificado de Desarrollo de Software

El Proceso Unificado de Desarrollo de Software es un marco de trabajo genérico que puede especializarse en una gran variedad de sistemas software, para diferentes áreas de aplicación, tipos de organizaciones, niveles de aptitud y tamaños de proyectos.

El Proceso Unificado tiene sus orígenes en los siguientes métodos:

- Método de Ericsson: desarrollado en el año 1967, este método modelaba el sistema entero como conjunto de bloques interconectados (en UML se lo conoce como subsistemas y se implementa mediante componentes). Los bloques de más bajo nivel se ensamblan formando subsistemas y luego sistemas. Se estudiaban los casos de negocio (hoy casos de uso) y el primer producto era una descripción de las actividades de la arquitectura del software basada en la comprensión de los requisitos más críticos.
- Método *Objectory*: este método se comienza a desarrollar en 1987 creado por Ivar Jacobson y su nombre se debe a la abreviatura de *Object Factory* que quiere decir “fábrica de objetos”. Este método usa la idea de casos de usos del método.
- Ericsson pero les da el nombre actual Método *Rational*: a finales de 1995 *Rational Software Corporation* compra a *Objectory* con lo cual nace el Proceso Unificado [Jacobson, 2000].

El Proceso Unificado está basado en componentes y, por tanto, según este método un sistema de software está formado por componentes de software interconectados mediante interfaces bien definidas. Se utiliza el Lenguaje Unificado de Modelado UML para preparar todos los esquemas de un sistema de software. Además, RUP se caracteriza por estar guiado por los casos de uso, entrado en la arquitectura y por ser iterativo e incremental:

- *Casos de uso*: es una técnica para capturar información de cómo un sistema o negocio trabaja, o de cómo se desea que trabaje.
- *Arquitectura*: se centra tanto en los elementos estructurales significativos del sistema, subsistemas, componentes y nodos, como en las colaboraciones que tienen lugar entre estos elementos mediante las interfaces. Los modelos que representan la arquitectura son: modelo de análisis, de diseño, de despliegue, de implementación y de casos de uso.
- *Iterativo e incremental*: el ciclo de vida iterativo se basa en la evolución de prototipos ejecutables que se muestran a los usuarios y clientes. En cada iteración los prototipos se van completando y se obtienen nuevos requisitos o se modifican los actuales.

### 2.2.2.1 Ciclo de vida del Proceso Unificado

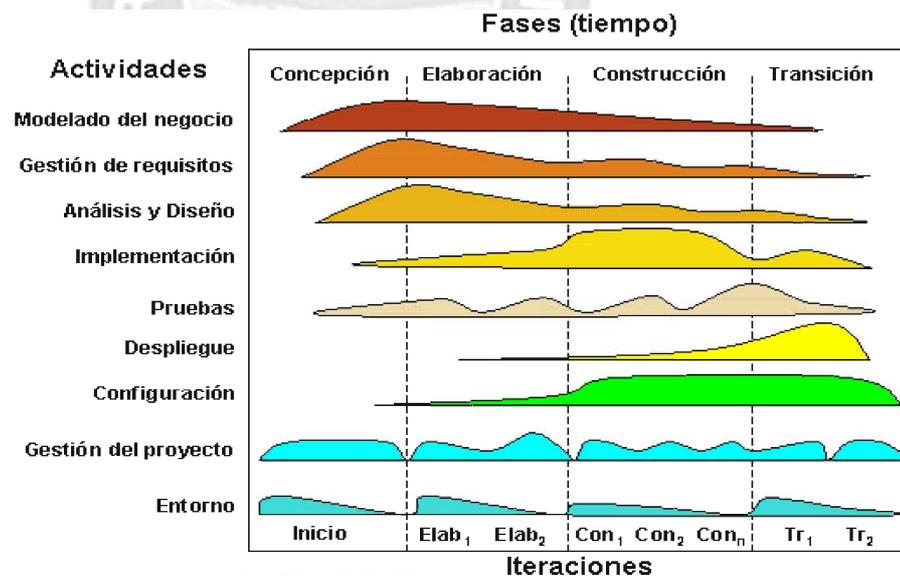


Figura 2.7: Ciclo de vida del Proceso Unificado. [Booch, 2000].

- **Iniciación.** Durante esta fase se desarrolla una descripción del producto final a partir de una buena idea, y se presenta el análisis de negocio para el producto. A lo largo de la misma se deben establecer las principales

funciones del sistema para los usuarios más importantes, la arquitectura del sistema a grandes rasgos y el plan de proyecto con una aproximación del coste del producto.

- **Elaboración.** Durante esta fase se especifica en detalle la mayoría de los casos de uso del producto y se diseña la arquitectura del sistema. Al final de esta fase el responsable del proyecto estará en condiciones de planificar las actividades y estimar los recursos necesarios para llevar a cabo el proyecto.
- **Construcción.** En esta fase la línea base de la arquitectura obtenida como producto durante la fase anterior crece hasta convertirse en el sistema completo.
- **Transición.** Cubre el periodo durante el cual el producto se convierte en versión beta. Conlleva actividades como la formación al cliente, la asistencia, resolución de incidencia y clasificación de aquella que justifica una versión del producto.

### 2.3 UML

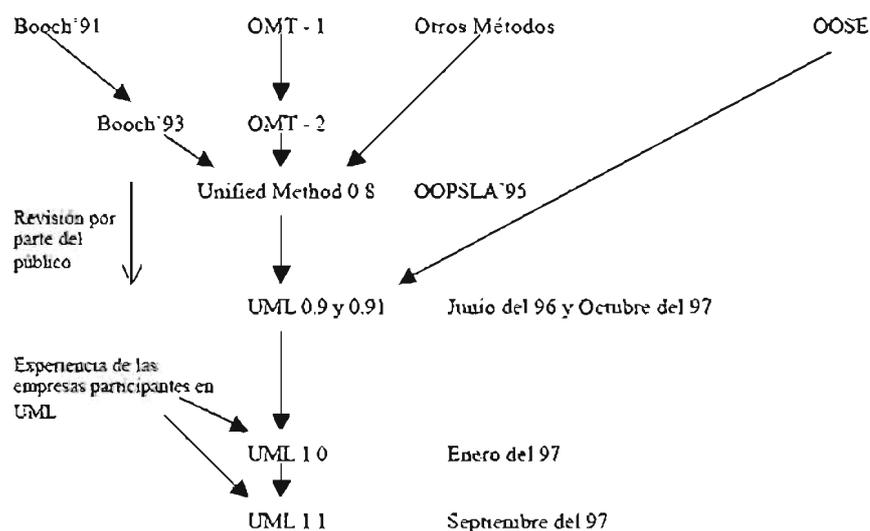


Figura 2.8: Historia de UML

UML (Unified Modeling Language) es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema software orientado a objetos. Se ha convertido en el estándar de la industria, debido a que ha sido impulsado por los autores de los tres métodos más usados de orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh. Estos autores fueron contratados por la empresa Rational Software Co. para crear una notación unificada en la que basar la construcción de sus herramientas CASE. En el proceso de creación de UML han participado, no obstante, otras empresas de gran peso en la industria como Microsoft, Hewlett-Packard, Oracle o IBM, así como grupos de analistas y desarrolladores.

Esta notación ha sido ampliamente aceptada debido al prestigio de sus creadores y debido a que incorpora las principales ventajas de cada uno de los métodos particulares en los que se basa (principalmente Booch, OMT y OOSE). UML ha puesto fin a las llamadas “guerras de métodos” que se han mantenido a lo largo de los 90, en las que los principales métodos sacaban nuevas versiones que incorporaban las técnicas de los demás. Con UML se fusiona la notación de estas técnicas para formar una herramienta compartida entre todos los ingenieros software que trabajan en el desarrollo orientado a objetos.

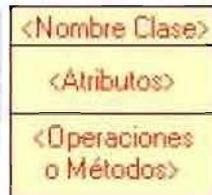
Uno de los objetivos principales de la creación de UML era posibilitar el intercambio de modelos entre las distintas herramientas CASE orientadas a objetos del mercado. Para ello era necesario definir una notación y semántica común.

### **2.3.1 Elementos**

Son bloques básicos de construcción orientados a objetos de UML. Se utilizan para escribir modelos bien formados. Existen cuatro tipos de elementos:

- Elementos estructurales. Son los nombres de los modelos UML. En su mayoría son las partes estáticas de un modelo, y representan cosas que son conceptuales o materiales. En total hay 7 tipos de elementos estructurales.

- **Clase:** Describe un conjunto de objetos con características y comportamiento idéntico (atributos, operaciones, relaciones y semántica).



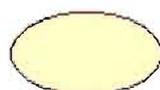
**Interfaz:** Es una colección de operaciones que especifican un servicio de una clase o componente, describe el comportamiento visible externamente de ese elemento.



**Colaboración:** Una colaboración define una interacción y es una sociedad de roles y otros elementos que colaboran para proporcionar un comportamiento cooperativo mayor que la suma de los comportamientos de sus elementos.



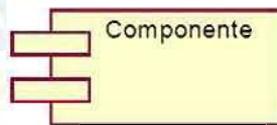
**Caso de uso:** Denota un requerimiento solucionado por el sistema. Es una descripción de un conjunto de secuencias de acciones que un sistema realiza y que produce un resultado.



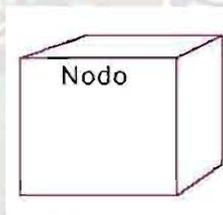
**Clase activa.** Es una clase cuyos objetos tienen uno o más procesos o hilos de ejecución y pueden dar origen a actividades de control. Una clase activa es igual

que una clase, excepto en que sus objetos representan elementos cuyo comportamiento es concurrente con otros elementos.

**Componente:** Un componente es una parte física y reemplazable de un sistema que conforma con un conjunto de interfaces y proporciona la implementación de dicho conjunto. Representa el empaquetamiento físico de diferentes elementos lógicos, como clases, interfaces y colaboraciones.



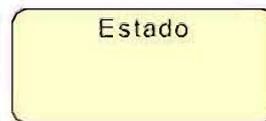
**Nodo:** Es un elemento físico que existe en tiempo de ejecución y representa un recurso computacional, que por lo general dispone de algo de memoria y, con frecuencia, capacidad de procesamiento. Un conjunto de componentes puede residir en un nodo y puede también migrar de un nodo a otro.



**Interacción:** Es un comportamiento que comprende un conjunto de mensajes intercambiados entre un conjunto de objetos, dentro de un contexto particular, para alcanzar un propósito específico. El comportamiento de una sociedad de objetos o una operación individual puede especificarse con una interacción. Una interacción involucra muchos otros elementos, incluyendo mensajes, secuencias de acción y enlaces.



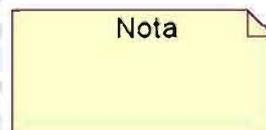
**Máquina de estados:** Una máquina de estados es un comportamiento que especifica las secuencias de estados por las que pasa un objeto o una interacción durante su vida en respuesta a eventos, junto con sus reacciones a estos eventos.



**Paquete:** Un paquete es un mecanismo de propósito general para organizar elementos en grupos. Los elementos estructurales, los elementos de comportamiento, e incluso otros elementos de agrupación pueden incluirse en un paquete, son elementos de agrupación básicos con los cuales se puede organizar un modelo UML.



**Nota:** Una nota es simplemente un símbolo para mostrar restricciones y comentarios junto a un elemento o una colección de elementos.



### 2.3.2 Relaciones

Las relaciones en UML son bloques básicos de construcción para relaciones, se utilizan para escribir modelos bien formados. Existen cuatro tipos de relaciones en UML.

- **Dependencia:** Una dependencia es una relación semántica entre dos elementos, en la cual un cambio a un elemento (llamado elemento independiente) puede afectar a la semántica del otro elemento (elemento dependiente).



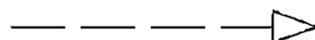
- **Asociación:** Es una relación estructural que describe un conjunto de enlaces, los cuales son conexiones entre objetos. De manera gráfica una asociación se representa como una línea continua, posiblemente dirigida, que a veces incluye una etiqueta, y con frecuencia otros adornos, como la multiplicidad y los nombres de rol.



- **Generalización:** Una generalización es una relación de especialización y/o generalización en la que los objetos del elemento especializado, o hijo, puede sustituir a los objetos del elemento general (padre). Así, el hijo comparte el comportamiento y la estructura del padre.



- **Realización:** Es una relación semántica entre clasificadores, en donde un clasificador especifica un contrato que otro clasificador garantiza que cumplirá.



### 2.3.3 Diagramas

Un diagrama es la representación gráfica de un conjunto de elementos, visualizado la mayoría de las veces como un grafo conexo de nodos (elementos) y arcos (relaciones). Estos diagramas se dibujan para visualizar un sistema desde diferentes perspectivas, de forma que un diagrama es una proyección de un sistema. Un diagrama representa una vista resumida de los elementos que constituyen un sistema. UML incluye nueve diagramas:

- **Diagrama de clases:** Un diagrama de clases muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Estos diagramas son los diagramas más comunes en el modelado de sistemas orientados a objetos, estos diagramas cubren la vista de diseño estática de un sistema.
- **Diagrama de objetos:** Muestra un conjunto de objetos y sus relaciones, representan instantáneas de instancias de los elementos encontrados en los diagramas de clases.
- **Diagrama de casos de uso:** Un diagrama de Casos de Uso muestra las distintas operaciones que se esperan de una aplicación o sistema y cómo se relaciona con su entorno.
- **Diagrama de secuencia:** Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo. Esta descripción es importante porque puede dar detalle a los casos de uso, aclarándolos al nivel de mensajes de los objetos existentes, como también muestra el uso de los mensajes de las clases diseñadas en el contexto de una operación.

- **Diagrama de colaboración:** Un diagrama de colaboración es una forma de representar interacción entre objetos, alterna al diagrama de secuencia. A diferencia de los diagramas de secuencia, pueden mostrar el contexto de la operación (cuáles objetos son atributos, cuáles temporales) y ciclos de ejecución.
- **Diagrama de estados.** Muestra el conjunto de estados por los cuales pasa un objeto durante su vida en una aplicación, junto con los cambios que permiten pasar de un estado a otro. Un estado Identifica un periodo de tiempo del objeto (no instantáneo) en el cual el objeto está esperando alguna operación, tiene cierto estado característico o puede recibir cierto tipo de estímulos.
- **Diagrama de actividades.** Un diagrama de actividades es un caso especial de un diagrama de estados que muestra el flujo de actividades dentro de un sistema. Todos los estados son estados de acción (identifican qué acción se ejecuta al estar en él) y casi todas las transiciones son enviadas al terminar la acción ejecutada en el estado anterior. Puede dar detalle a un caso de uso, un objeto o un mensaje en un objeto. Sirven para representar transiciones internas, sin hacer mucho énfasis en transiciones o eventos externos.
- **Diagrama de componentes:** Un diagrama de componentes muestra las dependencias lógicas entre componentes software, sean éstos componentes fuentes, binarios o ejecutables. Los componentes software tienen un tipo, que indica si son útiles en tiempo de compilación, enlace o ejecución. Se consideran en este tipo de diagramas solo tipos de componentes. Instancias específicas se encuentran en el diagrama de ejecución.

- **Diagrama de despliegue:** Muestra la configuración de nodos de procesamiento en tiempo de ejecución y los componentes que residen en ellos. Los diagramas de despliegue cubren la vista de despliegue estática de una arquitectura.

## 2.4 CALIDAD DEL SOFTWARE

La calidad de un producto software viene determinada por la satisfacción de unas exigencias de usuario. Dichas exigencias se traducen en la existencia de unas necesidades externas, que se logran, a su vez, a través de un proceso de desarrollo en el que se consideran unos requisitos de calidad internos. Posteriormente, el logro de unos y otros requisitos se contrasta a través de respectivos procesos de *verificación* y *validación*. Con la verificación se comprueba que el producto se está construyendo adecuadamente (punto de vista interno), y con la validación que se está desarrollando un producto software adecuado con las necesidades del usuario (punto de vista externo).

La Calidad de Software para Pressman (2002) es “la concordancia con los requisitos funcionales y de rendimiento establecidos con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado de forma profesional”.

### 2.4.1 Características de calidad

Se dice que cierto software tiene calidad si cumple 6 criterios descritos en la norma ISO 9126 que a continuación mencionaremos:

- **Funcionalidad:** Es el conjunto de atributos que permiten que un sistema provea ciertas propiedades y funciones específicas en otras palabras es el grado en que

las necesidades asumidas o descritas se satisfacen. Estas se subdividen de la siguiente forma:

- Adecuidad
- Exactitud
- Interoperabilidad
- Seguridad
- Conformidad de la funcionalidad

- **Fiabilidad:** Es el grado en que el sistema responde bajo las condiciones definidas durante un intervalo de tiempo dado. Estas se subdividen de la siguiente forma:

- Madurez.
- Tolerancia a Fallos.
- Recuperabilidad.

- **Usabilidad:** La usabilidad determina el grado de esfuerzo para que un usuario aprenda y use cierto software, que tan productivos son los usuarios que trabajan con ese sistema y cuanta ayuda necesitaran. Estas se subdividen de la siguiente forma:

- Aprendibilidad.
- Comprensibilidad.
- Operabilidad.
- Atractividad.

- **Eficiencia:** Es el conjunto de características que determinan la relación entre el nivel de rendimiento del software y el número de recursos usados, bajo ciertas condiciones dadas. Estas se subdividen de la siguiente forma:

- Comportamiento Temporal.
- Utilización de recursos.

- **Mantenibilidad:** Es el esfuerzo requerido para implementar cambios. Estas se subdividen de la siguiente forma:

- Analizabilidad.

- Cambiabilidad.
  - Estabilidad.
  - Facilidad de Prueba.
- 
- **Portabilidad:** Es el conjunto de características que determinan la capacidad del software para ser transferido de un entorno de operación a otro. Este se subdividen de la siguiente forma:
    - Adaptabilidad.
    - Instalabilidad.
    - coexistencia.
    - Reemplazabilidad.





## **CAPÍTULO III DESARROLLO DEL PROYECTO**

## CAPÍTULO III DESARROLLO DEL PROYECTO

### 3.1 ANÁLISIS DE REQUERIMIENTOS

Para el análisis de requerimiento fue de gran ayuda las entrevistas que tuve con el personal de control de asistencia y planillas de pago.

Los requisitos básicos que se tomaran encuentra para el desarrollo del proyecto son:

- Apoyarse en la herramienta de hardware como lo son los lectores biométricos de huella digital para optimizar el control de asistencia.
- El sistemas de información a implementar debe dar solución a problemas del control de personal y planillas de pago, deben cumplir con la política y perfil adoptados por ella.
- Las políticas de desarrollo de software en la actualidad y en el caso preciso de este proyecto se enmarcan en plataformas de software VB.NET y base de datos sql\_server.
- El sistema debe permanecer activo 5 días a la semana, durante todo el año (10 Hrs. Máximo por día).
- Los datos de asistencia debes ser capturados automáticamente a través del sistema biométrico.
- Las planillas de pago debe ser generadas automáticamente sin la necesidad de volver a introducir los datos asistencia.
- El sistema debe permitir adaptarse a un posible crecimiento de sus servicios y usuarios para satisfacer las demandas de rendimiento cada vez mayores.

- Adicionalmente el sistema se podrá usar como un medio de información al empleado.

## **3.2 ENCONTRAR ACTORES Y CASOS DE USO**

### **3.2.1 Encontrar actores**

Los actores encontrados en el sistema son los que se mencionan a continuación:

- Administrador
- Empleado

### **3.2.2 Encontrar casos de uso**

- Asignar horarios de trabajo
- Registrar empleado
- Validar usuario
- Registrar asistencia
- Historial de asistencia
- Reporte planillas de pago
- Reporte personal
- Reporte subsidios
- Reporte inasistencia
- Reporte permisos
- Reporte multas por retraso y/o faltas
- Asignar políticas de asistencia
- Planificar permisos
- Comisión

### 3.2.3 Priorizar casos de uso

La política aplicada para priorizar los casos de uso se basa en la elección, en primer lugar, de los casos de uso más importantes para el sistema, que son los que se detallan a continuación:

- Registrar Empleado
- Reportes
- Controlar asistencia
- Generar planillas de pago
- Planificar ausencia
- Imprimir

### 3.3 MODELADO DEL NEGOCIO

Los subsistemas en los que se ha dividido la Unidad a nivel de abstracción es el siguiente:



### Modelo de Casos de Uso del Negocio



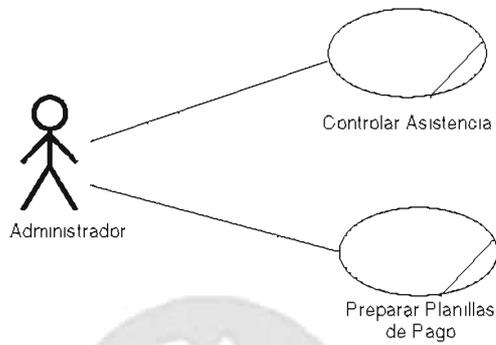


Figura 3.2: Modelo de Casos de Uso del Negocio

### Modelo de objetos de Marcar Asistencia

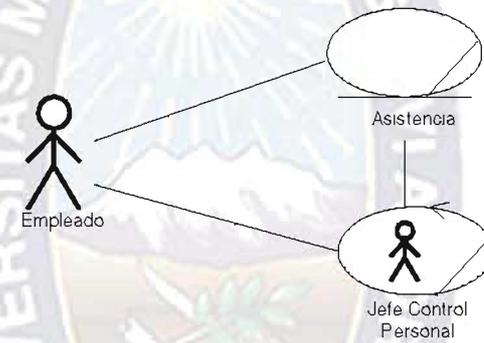


Figura 3.3: Modelo de objetos de Marcar Asistencia

### Modelo de objetos de Controlar Asistencia

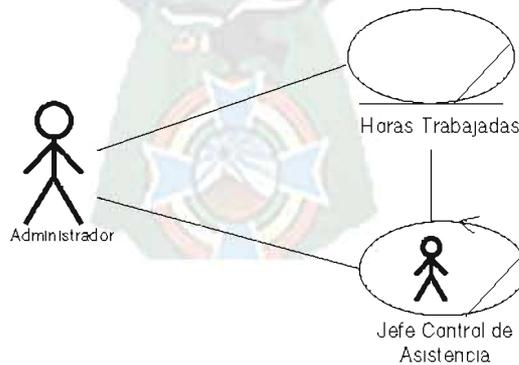


Figura 3.4: Modelo de objetos de Controlar Asistencia

### Modelo de objetos de Preparar Planillas de Pago

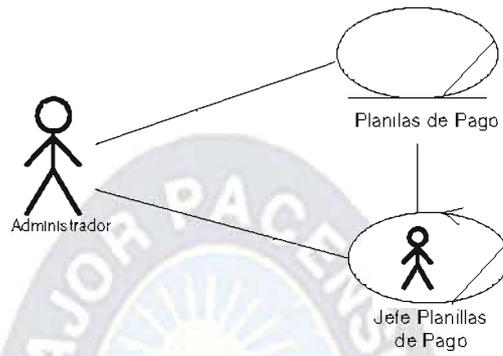


Figura 3.5: Modelo de objetos de Preparar Planillas de Pago

### 3.4 CASOS DE USO

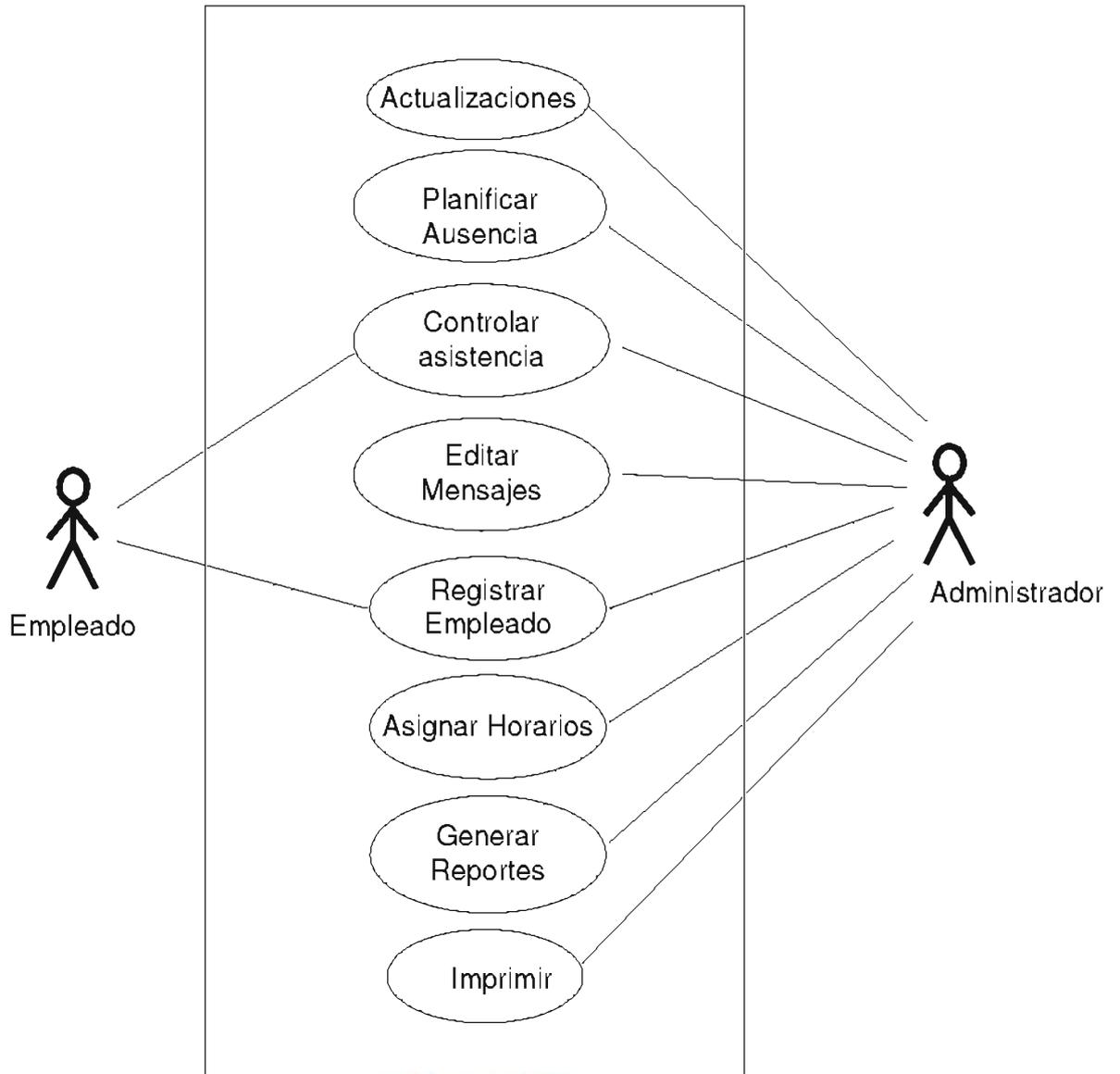


Figura 3.6: Diagrama Casos de Uso  
"Sistema Biométrico de Control de Asistencia y Planillas de Pago"

### 3.4.1 Detallar casos de uso

#### Caso de uso Controlar Asistencia



Figura 3.7: Caso de uso Marcar Asistencia

|   |  |
|---|--|
| <b>Caso de uso:</b> Capturar Datos Huella Digital   |  |
| <b>Actor:</b> Empleado  |  |
| <b>Descripción:</b> Permite obtener datos de la huella digital para su posterior comparación.                     |  |
| <b>Activación:</b> Se activa cuando el Empleado imprime su huella digital en el scanner biométrico.               |  |
| <b>Curso Normal</b>   | <b>Curso Alternativo</b>   |
| <b>1</b> El Empleado imprime su huella dactilar por medio del lector biométrico para su marcado de entrada/salida | <b>1.1</b> En caso de que no este registrado su huella, introduce su código en una casilla alternativa |
| <b>Precondiciones:</b> Empleado para el marcado de asistencia debe estar previamente registrado.                  |  |
| <b>Postcondiciones :</b> No se permite duplicidad de marcado.   |  |
| <b>Observaciones y datos:</b>   |  |

|   |   |
|---|---|
| <b>Caso de uso:</b> Marcar Asistencia   |   |
| <b>Actor:</b> Empleado<br>Administrador   |   |
| <b>Descripción:</b> Permite al empleado registrar su hora de entrada y salida.  |   |
| <b>Activación:</b> El caso de uso se activa después de un cierto tiempo después de haber hecho la impresión de huella digital en el scanner biométrico. |   |
| <b>Curso Normal</b>   |   |
| <b>1</b>  | Autenticación de funcionario  |
| <b>2</b>  | Registra datos de la hora y fecha de ingreso y salida marcada en la base de datos                           |
| <b>Curso Alternativo</b>  |   |
| <b>2.1</b>  | En casos de emergencia el Administrador puede ingresar los datos de hora de ingreso y salida por el teclado |
| <b>Precondiciones:</b> El empleado debe estar registrado previamente.   |   |
| <b>Postcondiciones :</b>  |   |
| <b>Observaciones y datos:</b>   |   |

|   |  |
|---|--|
| <b>Caso de uso:</b> Modificar Asistencia  |  |
| <b>Actor:</b> Administrador   |  |
| <b>Descripción:</b> Permite modificar el marcado de asistencia.                                     |  |
| <b>Activación:</b> El caso de uso se activa cuando el Administrador selecciona Modificar Asistencia |  |
| <b>Curso Normal</b>   |  |
| <b>1</b>  | Selecciona funcionario                   |
| <b>2</b>  | Seleccionar hora de marcado a modificar. |
| <b>3</b>  | Modificar hora de marcado                |
| <b>Curso Alternativo</b>  |  |
| <b>Precondiciones:</b> El empleado debe existir en la base de datos.                                |  |
| <b>Postcondiciones :</b> Modificar asistencias que se soliciten.                                    |  |
| <b>Observaciones y datos:</b>   |  |

|  |   |
|--|---|
| <b>Caso de uso:</b> Visualizar Mensajes  |   |
| <b>Actor:</b> Empleado   |   |
| <b>Descripción:</b> Permite informar a los empleados de algún acontecimiento, felicitaciones, llamadas de atención ,etc. |   |
| <b>Activación:</b> El caso de uso se activa cuando en el momento que el empleado hace su marcado.                        |   |
| <b>Curso Normal</b>  | <b>Curso Alternativo</b>  |
| 1  | Se visualiza automáticamente cuando el empleado hace el marcado de asistencia |
| <b>Precondiciones:</b> El mensaje debe ser editado anticipadamente.  |   |
| <b>Postcondiciones :</b> Visualizar siguiente mensaje.   |   |
| <b>Observaciones y datos:</b>  |   |

|   |                                       |
|---|---------------------------------------|
| <b>Caso de uso:</b> Adicionar Asistencia  |                                       |
| <b>Actor:</b> Administrador   |                                       |
| <b>Descripción:</b> Permite adicionar el marcado de asistencia para los empleados directamente por el teclado en casos de emergencia. |                                       |
| <b>Activación:</b> El caso de uso se activa cuando el Administrador selecciona Adicionar Asistencia                                   |                                       |
| <b>Curso Normal</b>   | <b>Curso Alternativo</b>              |
| 1   | Seleccionar empleado.                 |
| 2   | Introducir horas de entrada o salida. |
| 3   | Registrar horas de entrada o salida.  |
| <b>Precondiciones:</b> El empleado debe existir en la base de datos.  |                                       |
| <b>Postcondiciones :</b> Adicionar asistencias que se soliciten.  |                                       |
| <b>Observaciones y datos:</b>   |                                       |

### Caso de uso Registrar Empleado

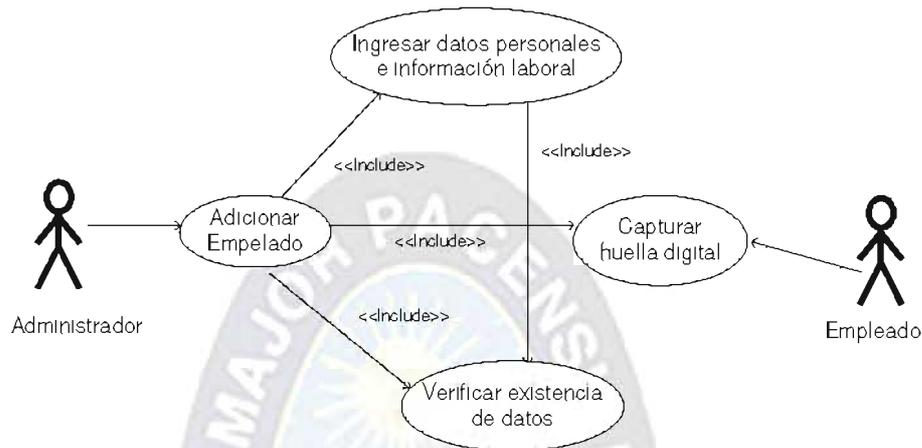


Figura 3.8: Caso de uso Registrar Empleado

|  |   |
|--|---|
| <b>Caso de uso: Adicionar Empleado</b>   |   |
| <b>Actor:</b> Administrador  |   |
| <b>Descripción:</b> EL empleado es registrado en el sistema. Durante el proceso de registro también se captura los datos de su huella digital con la ayuda del aparato biométrico. |   |
| <b>Activación:</b> El caso de uso se activa cuando el Administrador selecciona Adicionar Asistencia  |   |
| <b>Curso Normal</b>  | <b>Curso Alternativo</b>  |
| <b>1</b> El Administrador ingresa los datos del nuevo funcionario en el sistema.   | <b>1.1</b> Si el sistema detecta que ya existe el funcionario debe habilita una interfaz mensaje. |
| <b>2</b> El sistema confirma el ingreso de datos a través de validaciones de campos.   |   |
| <b>3</b> Envío de mensaje de bienvenida.   |   |
| <b>Precondiciones:</b> Validar permisos de acceso a creación de Empleado   |   |
| <b>Postcondiciones :</b>   |   |
| <b>Observaciones y datos:</b>  |   |

|  |   |
|--|---|
| <b>Caso de uso:</b> Capturar Huella digital  |   |
| <b>Actor:</b> Administrador<br>Empleado  |   |
| <b>Descripción:</b> El Administrador captura los datos generados por el scanner biométrico cuando el funcionario imprime su huella digital en el scanner biométrico. |   |
| <b>Activación:</b> El caso de uso se activa cuando el Administrador selecciona Capturar Huella digital   |   |
| <b>Curso Normal</b>  | <b>Curso Alternativo</b>  |
| <b>1</b>   | El funcionario imprime su huella digital en el escáner para ser registrado en la base de datos. |
| <b>2</b>   | El sistema confirma el ingreso de datos a través de validaciones.                               |
| <b>Precondiciones:</b> El dedo del Empleado debe estar en condiciones optimas para evitar errores de scanner de la huella digital.                                   |   |
| <b>Postcondiciones :</b>   |   |
| <b>Observaciones y datos:</b>  |   |



Figura 3.9: Caso de uso generar reportes

|   |  |
|---|--|
| <b>Caso de uso:</b> Generar Reportes  |  |
| <b>Actor:</b> Administrador   |  |
| <b>Descripción:</b> Muestra reportes.   |  |
| <b>Activación:</b> El caso de uso se activa cuando el Administrador selecciona Reportes.      |  |
| <b>Curso Normal</b>   | <b>Curso Alternativo</b>                                 |
| 1   | Seleccionar tipo de reporte                              |
| 2   | Cargar datos de reporte seleccionado.                    |
| 2.1   | En caso que no exista reporte habilita interfaz mensaje. |
| <b>Precondiciones:</b> No tiene   |  |
| <b>Postcondiciones :</b> El Administrador obtiene reportes que puede imprimir posteriormente. |  |
| <b>Observaciones y datos:</b>   |  |

### Caso de uso Planificar Ausencia

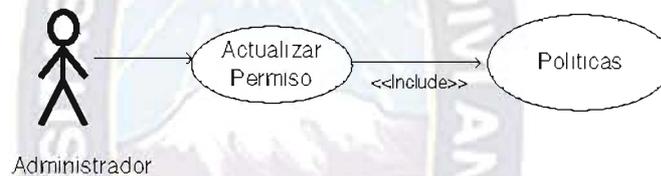


Figura 3.10: Caso de uso Planificar Ausencia

|  |                              |
|--|------------------------------|
| <b>Caso de uso:</b> Actualizar Permiso   |                              |
| <b>Actor:</b> Administrador  |                              |
| <b>Descripción:</b> Permite registrar anticipadamente permisos                                     |                              |
| <b>Activación:</b> El caso de uso se activa cuando el Administrador selecciona Actualizar permiso. |                              |
| <b>Curso Normal</b>  | <b>Curso Alternativo</b>     |
| 1  | Selecciona empleado          |
| 2  | Introducir datos del permiso |
| 3  | Actualizar permiso           |
| <b>Precondiciones:</b> No tiene  |                              |
| <b>Postcondiciones :</b>   |                              |
| <b>Observaciones y datos:</b>  |                              |

### Caso de uso Editar Mensajes

|  |                          |
|--|--------------------------|
| <b>Caso de uso:</b> Editar Mensajes  |                          |
| <b>Actor:</b> Administrador  |                          |
| <b>Descripción:</b> Permite editar mensajes anticipadamente para que el empleado se informe.   |                          |
| <b>Activación:</b> El caso de uso se activa cuando el Administrador selecciona Editar Mensaje. |                          |
| <b>Curso Normal</b>  | <b>Curso Alternativo</b> |
| 1  | Selecciona empleado      |
| 2  | Introducir mensaje       |
| 3  | Actualizar               |
| <b>Precondiciones:</b> No tiene  |                          |
| <b>Postcondiciones :</b>   |                          |
| <b>Observaciones y datos:</b>  |                          |

### Caso de uso Actualizaciones



Figura 3.11: Caso de uso Actualizaciones

|  |                                  |
|--|----------------------------------|
| <b>Caso de uso:</b> Actualizaciones  |                                  |
| <b>Actor:</b> Administrador  |                                  |
| <b>Descripción:</b> Permite actualizar diferentes tablas.                                  |                                  |
| <b>Activación:</b> El caso de uso se activa cuando el Administrador selecciona Actualizar. |                                  |
| <b>Curso Normal</b>  | <b>Curso Alternativo</b>         |
| 1  | Selecciona actualización         |
| 2  | Selecciona elemento a actualizar |
| 3  | Actualizar                       |
| <b>Precondiciones:</b> No tiene  |                                  |
| <b>Postcondiciones :</b>   |                                  |
| <b>Observaciones y datos:</b>  |                                  |

### 3.4 ANÁLISIS Y DISEÑO

#### Modelo de datos: Modelo Relacional

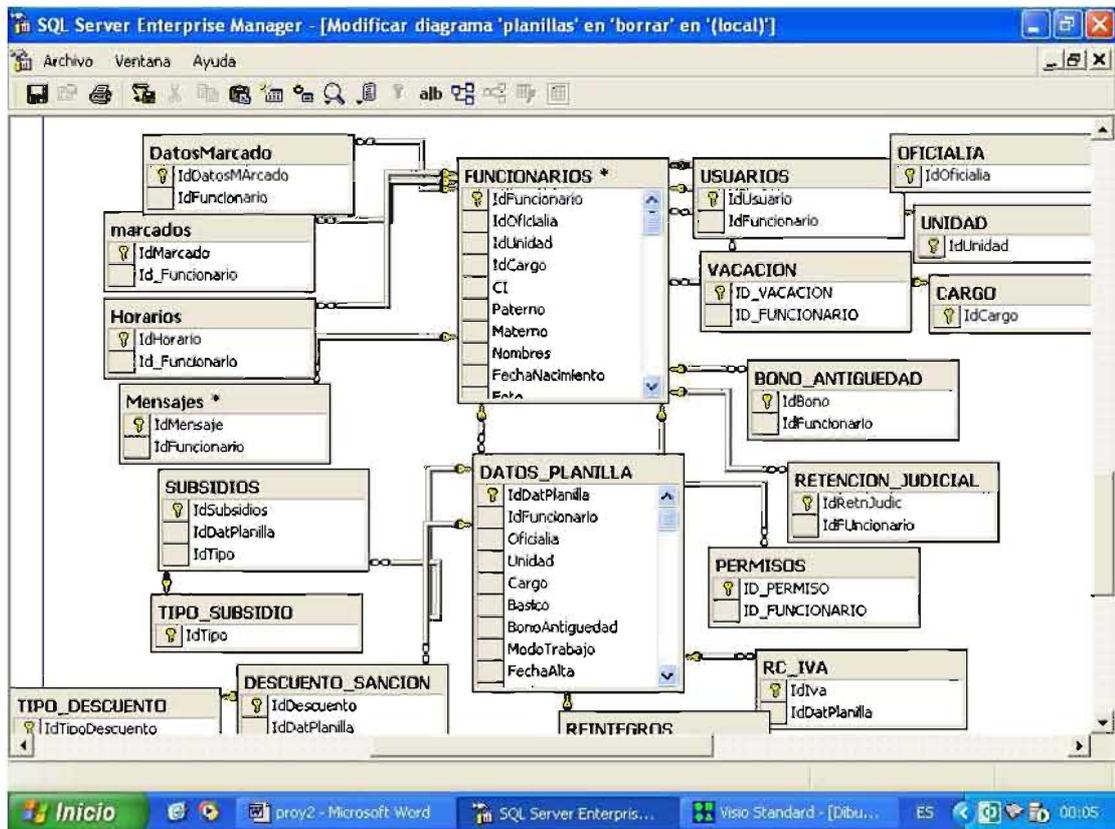


Figura 3.12: Modelo Relacional  
“Sistema Biométrico de Control de Asistencia y Planillas de Pago”

## 3.5 IMPLEMENTACIÓN

### 3.5.1 Diagrama global de paquetes

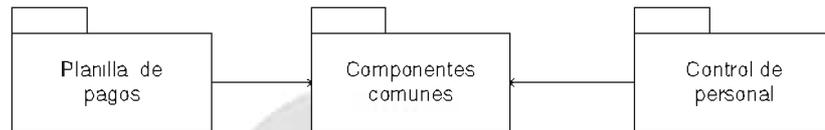


Figura 3.13: Diagrama global de paquetes

### 3.5.2 Diagrama de componentes

- Diagrama de componentes comunes

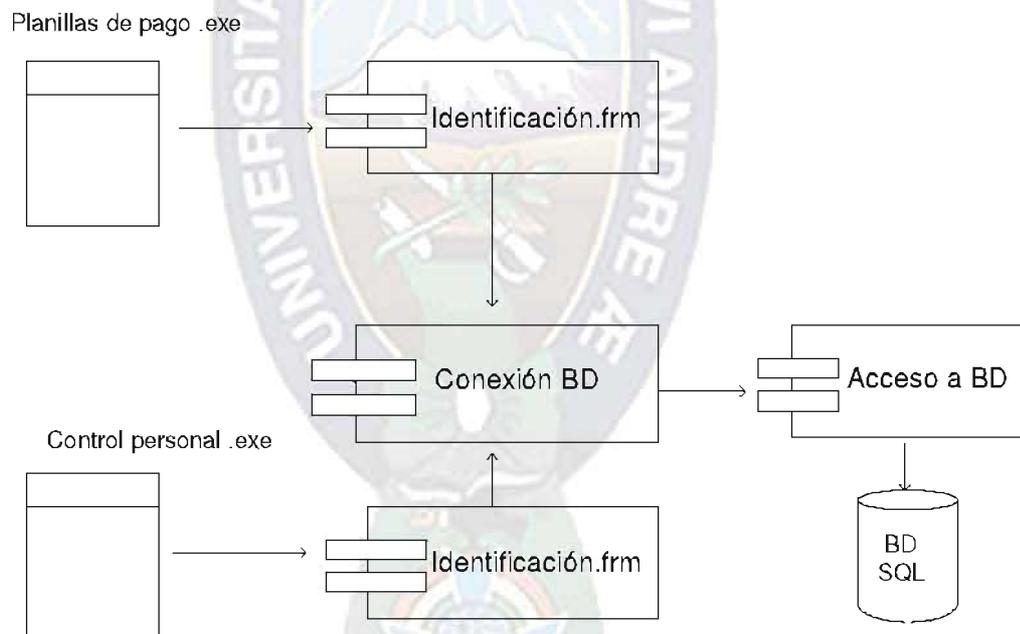


Figura 3.14: Diagrama de componentes comunes

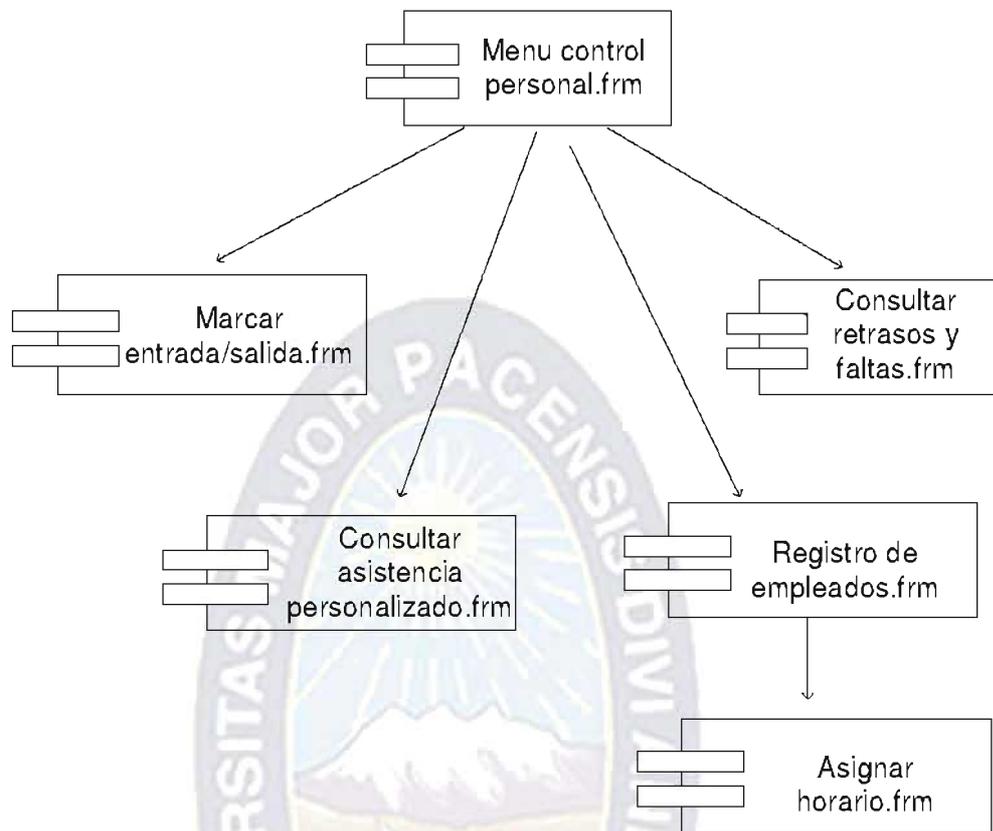


Figura 3.15 Diagrama de componentes control de personal

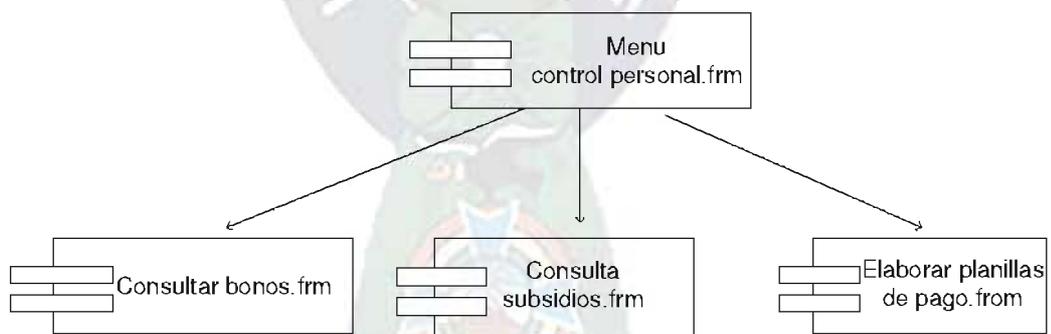


Figura 3.16: Diagrama de componentes planilla de pagos

### 3.5.3 Diagrama de despliegue

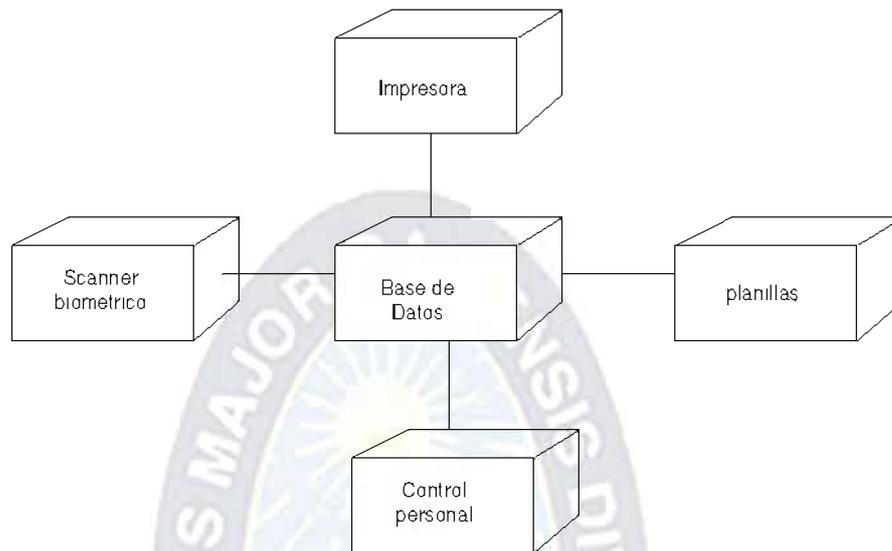


Figura 3.17: Diagrama de despliegue

### 3.5.4 Interfaces graficas de usuario

Identificación



Marcar asistencia



Menú principal (acceso restringido)



## Registro del empleado

**Funcionarios**

Datos (Requeridos) Foto (Opcional)

**Nombre(s)** Egberto  
**Apellido Paterno** Tola  
**Apellido Materno** Flores  
**Fecha Nacimiento** 02/02/1980  
**Cedula de Identidad** 4254149  
**Oficialia** Obras de medio ambiente  
**Unidad** Sistemas  
**Cargo** Tecnico

**Item** 425  
**Estatus** Alta  
**Fecha Alta** 02/02/04  
**Modo de Trabajo** Item

| Item_o_Contr | Paterno | Materno | Nombres | CI      | FechaNacimi | ModoTrabajo | Fecha    |
|--------------|---------|---------|---------|---------|-------------|-------------|----------|
| 425          | Tola    | Flores  | Egberto | 4254149 | 02/02/1980  | Item        | 02/02/04 |
| 659          | Losa    | Peres   | Juan    | 6598758 | 11/06/2007  | Item        | 02/02/07 |
| 859          | Lecoña  | Rosales | Raul    | 8597845 | 11/07/1980  | Item        | 02/02/07 |
| 859          | Lecoña  | Rosales | Raul    | 8597845 | 11/07/1980  | Item        | 02/02/07 |

## Descuentos

**descuentos\_Sancion**

**Tipo de Descuento** Llamada de atencion  
**Sanción** 50  
**Observaciones (Opcional)**  
**Mes Descuento** 1  
**AñoDescuento** 2007

**Nombre (s)** Egberto  
**Apellido Paterno** Tola  
**Apellido Materno** Flores  
**CI** 4254149

| Paterno | Materno | Nombres | CI      | Mes | Año  | Sancion | Ob | IdFunci |
|---------|---------|---------|---------|-----|------|---------|----|---------|
| Tola    | Flores  | Egberto | 4254149 | 1   | 2007 | 50      |    | 1000    |
| Lecoña  | Rosales | Raul    | 8597845 | 2   | 2007 | 230     |    | 1002    |
| Lecoña  | Rosales | Raul    | 8597845 | 2   | 2007 | 200     |    | 1002    |

\*

## Subsidios

subsidios

Fecha de Presentación: 05/03/2007  
 Fecha de Nacimiento: 14/05/2007  
 Tipo de Subsidio: Prenatal

Mes de Pago: 4  
 Año Pago: 2007  
 Nro de Paquete: 1

Observaciones (Opcional):

Nombre (s): Raul  
 Paterno: Lecoña  
 Materno: Rosales  
 CI: 8597845

|   | Paterno | Materno | Nombres | CI      | FechaPresent | FechaNacimi | MesPago | AñoPago | NroP. |
|---|---------|---------|---------|---------|--------------|-------------|---------|---------|-------|
| ▶ | Lecoña  | Rosales | Raul    | 8597845 | 05/03/2007   | 14/05/2007  | 4       | 2007    | 1     |
| * |         |         |         |         |              |             |         |         |       |

## Reintegro

REINTEGRO

Mes Reintegro: 6  
 Año Reintegro: 2007  
 Monto: 200

Descripción:

Nombre (s): Juan  
 Paterno: Losa  
 Materno: Peres  
 CI: 6598758

|   | Paterno | Materno | Nombres | CI      | Mes | Año  | Monto | Descripcio |
|---|---------|---------|---------|---------|-----|------|-------|------------|
| ▶ | Losa    | Peres   | Juan    | 6598758 | 6   | 2007 | 200   |            |
|   | Lecoña  | Rosales | Raul    | 8597845 | 2   | 2007 | 200   |            |
| * |         |         |         |         |     |      |       |            |

Usuario

USUARIOS

Nuevo     Modificar     Eliminar      

Usuario:

Perfil:

Contraseña:

Repetir Contraseña:

Reporte de Planilla de pagos

ReportPlanilla

Fecha Inicio: 17/06/2007    Fecha Fin: 17/06/2007   

|   | ITEM  | PATERNO | MATERNO   | NOMBRE  | CEDULA  | ALTA      | BAJA      | CARGO       | BA: |
|---|-------|---------|-----------|---------|---------|-----------|-----------|-------------|-----|
| ▶ | 70001 | Tola    | Flores    | Egberto | 4951020 | 17-Apr-03 | 14-Jul-03 | AUXILIAR    | 100 |
|   | 70008 | Losa    | Peres     | Juan    | 2700730 | 12-May-03 | 08-Aug-03 | PERSONAL    | 100 |
|   | 70009 | Lecoña  | Rosales   | Raul    | 4361529 | 02-May-03 | 29-Jul-03 | AUXILIAR    | 110 |
|   | 70010 | Salinas | Rojas     | Pedro   | 3440501 | 12-May-03 | 31-May-03 | TECNICO III | 180 |
|   | 70011 | Flores  | Rodriguez | Santos  | 2652396 | 01-Jul-03 | 27-Sep-03 | SECRETARI   | 140 |
| * |       |         |           |         |         |           |           |             |     |

### 3.7 FACTORES DE CALIDAD ISO 9126

#### 3.7.1 Confiabilidad

La confiabilidad del sistema es directamente proporcional a la calidad de sus componentes

$$R_i(t) = e^{-\lambda_i t} \quad (3.1)$$

Donde :

$R(t)$  : Función de confiabilidad de un componente en el tiempo  $t$ .

$\lambda$  : Tasa constante de fallo.

$t$  : Período de operación de tiempo.

**Teorema 1.** Si  $n$  componentes, funcionan independientemente conectados en serie y el  $i$ -ésimo componente tiene confiabilidad  $R_i(t)$ , entonces la confiabilidad total esta dada por :

$$R(t) = R_1(t) * R_2(t) * R_3(t) \dots R_n(t) \quad (3.2)$$

**Teorema 2.** Si  $n$  componentes, funcionan independientemente y actúan en paralelo y el  $i$ -ésimo componente tiene confiabilidad  $R_i(t)$ , entonces la confiabilidad total esta dada por:

$$R(t) = 1 - [1 - R_1(t)] * [1 - R_2(t)] * [1 - R_3(t)] * \dots * [1 - R_n(t)] \quad (3.3)$$

Si se tiene el siguiente diagrama de transferencia:



Fig. 3.18: Diagrama de transferencia

Fuente: Elaboración propia

Luego obtenemos la siguiente tabla

| $R_i(t)$ | $\lambda$ | $t$ [hrs] | $e^{-\lambda * t}$ |
|----------|-----------|-----------|--------------------|
| $R_1(t)$ | 0.01      | 10        | 0.90               |
| $R_2(t)$ | 0.005     | 10        | 0.95               |
| $R_3(t)$ | 0.01      | 10        | 0.90               |
| $R_4(t)$ | 0.01      | 10        | 0.90               |
| $R_5(t)$ | 0.005     | 10        | 0.95               |
| $R_6(t)$ | 0.01      | 10        | 0.90               |

Tabla 3.1: Cálculo de confiabilidad

Fuente: Elaboración propia

Utilizando los teoremas 1 y 2 calculamos:

$$R_T(t) = \{ 1 - [1 - R_3(t)] * [1 - R_4(t)] * [1 - R_5(t)] \} * [R_1(t)] [R_2(t)] [R_6(t)]$$

$$R_T(t) = \{ 1 - [1 - 0.90] * [1 - 0.90] * [1 - 0.95] \} * 0.90 * 0.95 * 0.90$$

$$R_T(t) = 0.77$$

$$\text{Confiabilidad} = R_T(t) * 100 \% = 77\%$$

### 3.7.2 Funcionalidad

La ecuación de punto función permitirá su cálculo:

$$PF = \text{Cuenta Total} * (A + (E * \text{Sum} (F_i))) \quad (3.4)$$

Donde :

PF : Funcionalidad estimada del sistema.

Cuenta Total : Representa la suma de todas las entradas.

A : Representa la portabilidad subjetiva estimada de la confiabilidad.

E : Error estimado del sistema.

SUM (F<sub>i</sub>) : Valores de ajuste de complejidad

Primero identificamos los parámetros de medición

|                               |  |
|-------------------------------|--|
| Numero Entradas de Usuario    | Registro de funcionario.<br>Registro de subsidios.<br>Registro de descuento<br>Registro de reintegros.<br>Registro de RC-IVA<br>Registro de bonos<br>Registro de horario<br>Consultas. |
| Numero Salidas de Usuario     | Reportes de planillas de pago<br>Reportes marcado de asistencias<br>Reportes de retenciones impositivas<br>Reportes de sanción por faltas y atrasos                                    |
| Numero Peticiones del Usuario | Menú principal del sistema<br>Búsquedas de: Funcionarios, historial planillas, historial asistencia, actualizaciones.  |
| Numero de Archivos            | Funcionarios, Datos_planilla, Bonos, Reintegros, Retenciones, Descuentos,  |

|                               |   |
|-------------------------------|---|
|                               | Subsidios, Horarios, Marcado, Permisos, Mensajes, Rerg_Huellas, Tipo_Subsidio, Tipo_Descuento, Datos_Retenciones_Impositivas, . |
| Numero de interfaces externas | Disquetes, CD, información de respaldo, Backups   |

Tabla 3.2: Parámetros de medición

| Parámetro de medida             | Cuenta | Ponderación | Total      |
|---------------------------------|--------|-------------|------------|
| Numero de entradas de usuario   | 8      | 4           | 32         |
| Numero de salidas de usuario    | 4      | 5           | 20         |
| Numero de peticiones de usuario | 6      | 6           | 36         |
| Numero de archivos              | 15     | 7           | 105        |
| Numero de interfaces externas   | 4      | 5           | 20         |
| <b>Cuenta total</b>             |        |             | <b>213</b> |

Tabla 3.3: Calculo de cuenta total

Fuente: Elaboración propia

Los ajustes de complejidad se realizan de acuerdo a las siguientes respuestas:

| Importancia  | Escala        |            |          |       |               |          |
|--|---------------|------------|----------|-------|---------------|----------|
|  | 0%            | 20%        | 40%      | 60%   | 80%           | 100%     |
| Factor   | No influencia | Incidental | moderado | Medio | Significativo | Esencial |
|  | 0             | 1          | 2        | 3     | 4             | 5        |
| 1. ¿Requiere el sistema copias de seguridad y de recuperación fiables? |               |            |          |       |               | X        |
| 2. ¿Se requiere comunicación de datos?                                 |               |            |          | X     |               |          |
| 3. ¿Existen funciones de procesamiento distribuido?                    |               | X          |          |       |               |          |
| 4. ¿Es critico rendimiento?  |               | X          |          |       |               |          |

|   |  |   |   |   |   |   |
|---|--|---|---|---|---|---|
| 5. ¿Se ejecutara el sistema en un entorno operativo existente y fuente utilizado?   |  |   |   |   | X |   |
| 6. ¿Requiere el sistema entrada de datos interactivo?   |  |   |   | X |   |   |
| 7. ¿Requiere la entrada de datos interactivos que las transacciones de entrada se lleven a cabo sobre múltiples pantallas o variadas utilizado? |  | X |   |   |   |   |
| 8. ¿Se actualizan los archivos maestros de forma interactiva?   |  |   |   | X |   |   |
| 9. ¿Son complejas las entradas, las salidas o las operaciones?  |  |   | X |   |   |   |
| 10. ¿Es complejo el procesamiento interno?  |  |   | X |   |   |   |
| 11. ¿Se ha diseñado el código para ser reutilizable?  |  |   |   |   | X |   |
| 12. ¿Están incluidas en el diseño la conversión y la instalación?   |  |   |   | X |   |   |
| 13. ¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?  |  | X |   |   |   |   |
| 14. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?                                    |  |   |   |   |   | X |

$$\sum F_i = F_1 + F_2 + F_3 + F_4 + F_5 + F_6 + F_7 + F_8 + F_9 + F_{10} + F_{11} + F_{12} + F_{13} + F_{14}$$

$$\sum F_i = 5+3+1+1+4+3+1+3+2+2+4+3+1+5$$

$$\sum F_i = 38$$

Reemplazando en la ecuación (3.4) para un nivel de confianza del 65%

$$PF_{real} = \text{cuenta total} * (A + (E * \text{Sum}(F_i)))$$

$$PF_{real} = 213 * (0.65 + (0.01 * 38))$$

$$PF_{real} = 219.39$$

Ahora calculamos para un nivel de confianza del 100%

$$PF_{esperado} = \text{cuenta total} * (A + (E * \text{Sum}(F_i)))$$

$$PF_{esperado} = 213 * (1 + (0.01 * 38))$$

$$PF_{esperado} = 293.94$$

El porcentaje de funcionalidad será como sigue:

$$\% PF = PF_{\text{real}} / PF_{\text{esperado}} = 219.39 / 293.94 = 0.75$$

Por lo tanto se tiene una funcionalidad de 75%

### 3.7.3 Facilidad de uso

| Usuarios        | Facilidad de comprensión | Facilidad de aprendizaje | Facilidad de operación |
|-----------------|--------------------------|--------------------------|------------------------|
| Usuario 1       | 90%                      | 90%                      | 90%                    |
| Usuario 2       | 95%                      | 90%                      | 95%                    |
| Usuario 3       | 95%                      | 95%                      | 100%                   |
| <b>Promedio</b> | 93.33                    | 91.67                    | 95                     |

La facilidad de uso es 93%

### 3.7.4 Facilidad de mantenimiento

Para medir la facilidad de mantenimiento aplicamos la métrica de tiempo medio TMC.

$$TMC = TA + TD + TI + TP \quad (4.5)$$

Donde :

**TA:** Tiempo en el que se tarda analizar la petición de cambio.

**TD:** Tiempo empleado en diseñar una modificación adecuada.

**TI :** Tiempo necesario para implementar el cambio.

**TP:** Tiempo empleado en probar y distribuir el cambio a todos los usuarios.

### 3.7.4.1 Mantenimiento correctivo

| En el mejor de los casos [min]  | En el peor de los casos[hrs] |
|---------------------------------|------------------------------|
| TMC= 20 + 30 + 50 + 100         | TMC= 24 + 24 + 48 + 24       |
| TMC= 200 [min] $\cong$ 3.3[hrs] | TMC= 120 [hrs]               |

El promedio de facilidad de mantenimiento adaptivo es de 62 hrs.

### 3.7.4.2 Mantenimiento de mejora

| En el mejor de los casos [hrs] | En el peor de los casos[dias] |
|--------------------------------|-------------------------------|
| TMC= 5+ 5 + 8 + 12             | TMC = 15 + 30 + 25 + 10       |
| TMC=30                         | TMC= 80                       |

El promedio de facilidad de mantenimiento de mejora es 975 hrs.

### 3.7.5 Portabilidad

El grado de portabilidad esta dado por:

$$GP= 1- [CT/CRD]$$

Donde :

GP: Grado de portabilidad

CT costo de transportar

CRD Costo de Re-Desarrollo

Si  $GP > 0$ : La portabilidad es mas rentable que el re-desarrollo

Si  $GP = 1$ : La portabilidad es perfecta

Si  $GP < 0$ : El Re desarrollo es ,mas rentable que la portabilidad.

Realizando los cálculos tendremos:

$$GP= 1- [CT/CRD]$$

$$GP=1-[70\$/1500\$]$$

$$GP=0.95$$

Por lo tanto es portable en un 95 %



## **CAPÍTULO IV CONCLUSIONES Y RECOMENDACIONES**

## **CAPÍTULO IV CONCLUSIONES Y RECOMENDACIONES**

### **4.1 CONCLUSIONES**

El software de este proyecto cumplió con los requerimientos y objetivos desde su planeación.

Con este proyecto no se pretende en ningún caso, reemplazar al personal encargado del control de asistencia. Al contrario este sistema será un apoyo para las funciones que efectúa.

Con el “sistema biométrico de control de asistencia y planillas de pago” el control de asistencia será mas efectivo, minimizando el tiempo de emisión de reportes de control de personal y planillas de pago.

Las expectativas de sistemas biométricos en nuestro país son muy buenas.

### **4.2 RECOMENDACIONES**

Es necesario de un administrador del sistema que realice mantenimiento y apoyo técnico a los usuarios.

Para la elaboración del sistema se tomo en cuenta la Unidad de control de asistencia y Planilla de Pagos pero no la Unidad de Subsidios Familiares que también tiene procesos complejos que al final se integra a Planillas. Se recomienda realizar este modulo y integrarlo a este sistema que presento.

## GLOSARIO

**G.M.E.A.** Gobierno Municipal de El Alto

**USB:** Puerto serial de la computadora

**Biometría:** la biometría analiza y mide ciertas características de una persona para crear un identificador biométrico el cual puede ser almacenado en una base de datos y recuperado para su comparación.

**Huella dactilar:** es la representación de la morfología superficial de la epidermis de un dedo, estas huellas son diferentes en cada persona  
Identificar

**Sistemas Biométricos:** sistema automatizado que realiza labores de biometría. Es decir, un sistema que fundamenta sus decisiones de reconocimiento mediante una característica personal que puede ser reconocida o verificada de manera automatizada.

**Reloj mecánico marcador de tarjetas:** Es un reloj mecánico en donde se introduce una tarjeta, y este reloj estampa la hora actual.

**Marcar asistencias:** Registrar la hora de entrada o salida

## REFERENCIAS BIBLIOGRÁFICAS

[KENDALL, K.E(1999)] “Análisis y Diseño de Sistema”

Editorial PRENTICE HALL HISPANOAMÉRICA

[PRESSMAN (2002)] “Ingeniería Del software”

Editorial: Concepción Fernández Madrid España

[BOOCH (1998)] “ Análisis y Diseño Orientado a Objetos”

2da Edición

[RUMBAUGH (1996)] “Modelado y Diseño Orientado a Objetos”

Editorial PRENTICE may

[JACABOSON, Booch, G., Rumbaugh J.] “ El Proceso Unificado de

Desarrollo de Software”

## SITIOS WEB

- <http://www.creangel.com/uml/diagramas.php>
- <http://www.programacion.net/tutorial/uml/9/>
- [http://www.vico.org/aRecursosPrivats/TRAD\\_introUML.pdf](http://www.vico.org/aRecursosPrivats/TRAD_introUML.pdf)
- [http://www.vico.org/aRecursosPrivats/TRAD\\_introUML.pdf](http://www.vico.org/aRecursosPrivats/TRAD_introUML.pdf)
- <http://www.clikear.com/manuales/uml/>
- <http://www.exa.unicen.edu.ar/cursosposg/docs/2007/Desarrollo%20de%20Software%20con%20UML.pdf>
- [http://eisc.univalle.edu.co/materias/Material\\_Desarrollo\\_Software/UML\\_Historia\\_Intro.pdf](http://eisc.univalle.edu.co/materias/Material_Desarrollo_Software/UML_Historia_Intro.pdf)
- <http://www.cs.ualberta.ca/~pfiguero/soo/uml/implementacion01.html>
- <http://www.di.uniovi.es/~dflanvin/doctorado/pi-lanvin.pdf>
- <http://www.mcc.unam.mx/~cursos/Objetos/Omt/omt.html>
- <http://www.tecnociencia.es/monograficos/biometria/biometria2.html>
- <http://www.tecnociencia.es/monograficos/biometria/biometria3.html>
- <http://usuarios.lycos.es/oopere/uml.htm>
- <http://www.dccia.ua.es/dccia/inf/asignaturas/GPS/archivos/Uml.PDF>
- <http://www.lcc.uma.es/~av/Publicaciones/04/UMLProfiles-Novatica04.pdf>
- <http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/multiple-html/index.html>
- <http://www.clikear.com/manuales/uml/>
- <http://www.cienciadigital.es/hemeroteca/reportaje.php?id=83>



## **ANEXOS**

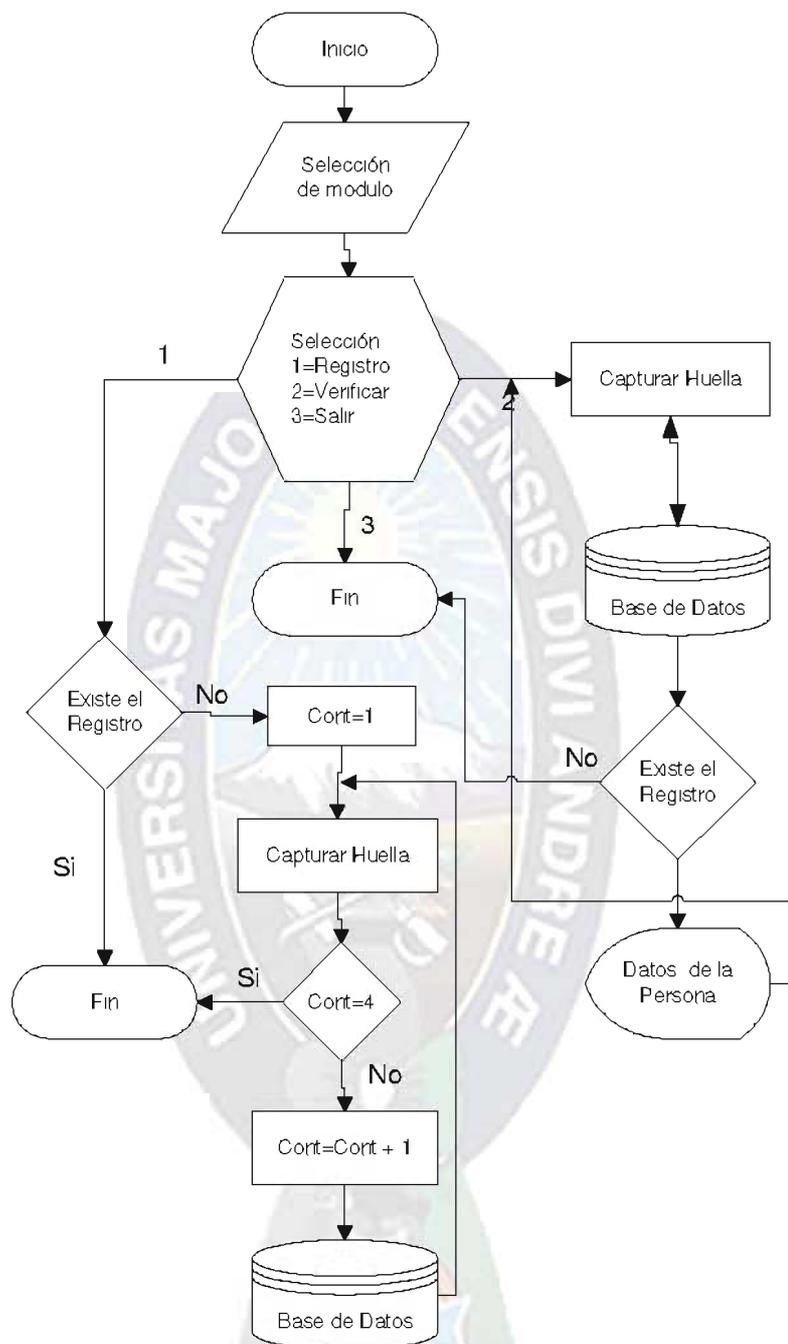
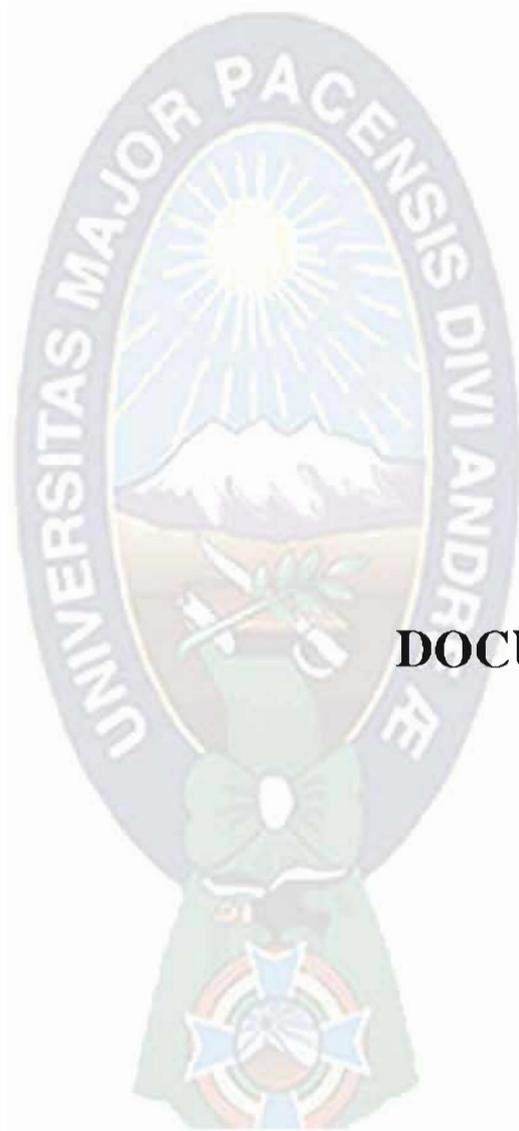


Diagrama de flujo del reconocimiento de huella dactilar



## DOCUMENTACIÓN

