

UNIVERSIDAD MAYOR DE SAN ANDRES
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA



PROYECTO DE GRADO
SISTEMA DE OFERTA Y DEMANDA PARA EL APOYO A LA TOMA
DE DECISIONES CON OLAP
S.O.D.A.T.D.
CASO: “SENAMHI”
PARA OPTAR AL TITULO DE LICENCIATURA EN INFORMÁTICA
MENCIÓN: INGENIERIA DE SISTEMAS INFORMÁTICOS

AUTOR : MARICEL LUCERO ALBERTO ALBERTO
TUTOR : LIC. GERMAN HUANCA TICONA
REVISOR : LIC. BRIGIDA CARVAJAL BLANCO

LA PAZ – BOLIVIA

2010

DEDICATORIA

A mi hermana Maribel Lucero Alberto Alberto por el apoyo y
consejos que me brindo siempre.

A mis padres Florinda y Eussy, por que los amo y admiro por
su paciencia, bondad y sabiduría.

A mi gran familia!

Gracias

AGRADECIMIENTOS

A Dios por iluminar mi camino y permitirme llegar hasta donde estoy.

Al Lic. Germán Huanca Ticona por guiarme de forma paciente, dándome consejos para concluir el proyecto y por la motivación constante que siempre me ha brindado.

A la Lic. Brígida Carvajal Blanco por haber sido parte de mi proyecto con las revisiones constantes del documento, por sus sugerencias que dieron forma al documento.

También debo agradecer a los docentes de la Carrera de Informática, por inculcarme sus conocimientos, sabiduría y valores durante el transcurso de mi vida universitaria.

Al Ministerio de Medio Ambiente y Agua (SENAMHI) al Jefe de la unidad Tecnología de Información y Comunicaciones al Ing. Antonio Terán Revilla y al Ing. Erick Pereyra por brindarme la información necesaria y al personal por la colaboración brindada.

Doy gracias a mis padres por su apoyo, comprensión y cariño incondicional que es parte de mi fortaleza.

Doy gracias a mi Hermana por su apoyo, comprensión y cariño incondicional que es parte de mi fortaleza.

Un agradecimiento muy especial de corazón a todos los amigos y amigas que formaron parte del proyecto, por su ayuda, paciencia y apoyo.

Agradezco también todos mis amigos y amigas de la Carrera de Informática y Arquitectura con quienes compartimos momentos inolvidables, gracias por su amistad y su apoyo.

Muchas Gracias.....

maricel_bloom@hotmail.com

RESUMEN

En la actualidad las Tecnologías de Comunicación, Sistemas de Información e Internet forman parte de una integración que beneficia a las organizaciones, fomentando la competencia para lograr mayores beneficios en una organización, lo que implica que cualquier persona para comunicarse, informarse, etc. Ya no necesita ir hasta donde se almacena la información requerida, basta con tener acceso al servidor de la institución donde puede realizar su consulta, de tal manera que se le facilita mucho la obtención de información, por tal motivo el Servicio Nacional de Meteorología e Hidrología SENAMHI tiene la necesidad de implementar un Sistema de oferta y demanda para el apoyo a la toma de decisiones que facilite las funciones que desempeñan los administrativos de la institución y la obtención de información oportuna y precisa .

De tal manera que el presente proyecto se orienta en solucionar los problemas de la Institución, que se constituye en Servicio Nacional de Meteorología e Hidrología que ofrece venta de certificaciones de datos meteorológicos e hidrológicos, certificación de servicios y la certificación de productos especializados en base a proformas elaboradas en la institución para los clientes que requieren la información.

Con la implementación del Sistema de Oferta y Demanda para el apoyo a la Toma de Decisiones con OLAP, se ayudará a desempeñar las funciones de los administrativos implicados en el área, para ello se considera los siguientes puntos importantes: Para el control de la demanda y la oferta se toma en cuenta el registro de los clientes (Demanda) que solicitan certificación el registro de los productos (oferta) y el registro de las proformas elaboradas en la institución.

Aplicando uno de los procesos de desarrollo más generales RUP, el cual está pensado para adaptarse a cualquier tipo de proyecto, se pretende modelar el sistema según las necesidades de los usuarios, para el desarrollo del sistema se utiliza JAVA, el servidor Mondrian para aplicar OLAP como gestor de base de datos POSTGRES

De esta forma se pretende ayudar a los funcionarios del *SENAMHI* a realizar un control sobre la oferta y demanda con la que cuenta, obteniendo información oportuna, confiable y fidedigna para poder realizar decisiones a favor de la institución.

ABSTRACT

Today Communication Technology, Information Systems and the Internet are part of an integration that benefits organizations by promoting competition to achieve greater benefits in an organization, which means that any person to communicate, learn, etc. You no longer need to go to where it stores the information required, just access the server from the institution where you can consult, so that it greatly facilitates the collection of information, for this reason the National Service of Meteorology and Hydrology SENAMHI has the need to implement a system of supply and demand to support decision-making to facilitate the administrative functions performed by the institution and to obtain timely and accurate information.

So that this project aims to solve the problems of the institution, which constitutes the National Service of Meteorology and Hydrology, offering for sale of certificates of meteorological and hydrological data, certification and certification services of specialized products based on made proformas in the facility for customers who require the information.

With the implementation of Supply and Demand System to support Decision Making with OLAP, will help carry out the administrative functions involved in the area, this will consider the following points: For the control of demand and supply is taken into account the enrollment of clients (Application) seeking registration certification of products (supply) and the registration of the institution made proformas.

Applying one of the broader development processes RUP, which is designed to suit any type of project aims to model the system as user needs, system development using Java, the server to implement OLAP Mondrian as a database manager POSTGRES

This form is intended to help officials make SENAMHI to control supply and demand that counts, obtaining timely, reliable and trustworthy to make decisions for the institution.

INDICE

CAPITULO I

1. Introducción.....	1
1.1. Antecedentes.....	2
1.1.1. Institucionales.....	3
1.1.2. Otros Proyectos.....	4
1.2. Planteamiento del Problema.....	5
1.3. Justificación.....	6
1.3.1. Justificación Tecnológica.....	6
1.3.2. Justificación Económica.....	6
1.3.3. Justificación Social.....	6
1.4. Objetivo.....	7
1.4.1. Objetivo General.....	7
1.4.2. Objetivo Especifico.....	7
1.5. Alcances.....	8
1.6. Herramientas.....	9

CAPITULO II

2. Marco Teórico.....	10
2.1. OLAP (Online Analytical Processing).....	10
2.2. Cubo OLAP.....	12
2.3. Data WareHouse.....	13
2.4. Modelo Estrella.....	14
2.5. Tabla de Hechos.....	14
2.6. Tabla Dimension.....	14
2.7. Mondrian.....	15
2.8. Jpivot.....	16
2.9. Java 2 Platform, Enterprise Edition (J2EE).....	17
2.10. Java Server Faces.....	18
2.11. Ajax.....	19

2.12.	PostgreSQL.....	20
2.13.	Lenguaje Unificado de modelado UML.....	20
2.14.	Proceso Unificado de Desarrollo de Software (RUP).....	21
2.15.	Las cuatro fases de Desarrollo del RUP.....	22
2.16.	Métricas de Calidad.....	27
2.17.	Clasificación De Las Métricas De Software.....	28
2.18.	Métricas De Software Orientadas a la Función.....	29
2.19.	ISO 9126.....	29
2.19.1.	Funcionalidad.....	30
2.19.2.	Fiabilidad.....	34
2.19.3.	Usabilidad.....	34
2.19.4.	Eficiencia.....	34
2.19.5.	Mantenibilidad.....	34
2.19.6.	Portabilidad.....	35

CAPITULO III

3.	Desarrollo del Sistema.....	36
3.1.	Introducción.....	36
3.1.1.	Análisis y situación actual.....	36
3.2.	Fase de Inicio.....	36
3.2.1.	Requerimientos del sistema.....	36
3.2.1.1.	Requerimientos Funcionales.....	37
3.2.1.2.	Requerimientos no Funcionales.....	37
3.2.2.	Descripción del Sistema.....	38
3.3.	Fase de Elaboración.....	39
3.3.1.	Modelado del Negocio.....	39
3.3.2.	Descripción de Actores.....	39
3.3.3.	Identificación de Casos de Uso.....	40
3.3.4.	Diagrama de clases.....	50
3.3.4.1.	Descripción De Tablas De Diagrama De Clases...	51

3.3.5. Diagrama de Secuencia.....	52
3.3.6. Diagrama de Estados.....	57
3.3.7. Diagrama de Colaboración.....	61
3.3.8. Cubo.....	63
3.4. Fase de Construcción.....	64
3.4.1. Diagrama de Componentes.....	65
3.4.2. Diagrama de Despliegue.....	66
3.4.3. Arquitectura del Sistema.....	67
3.4.4. Implementación.....	67
3.4.4.1. Ventanas del Sistema.....	67
3.5. Fase de Transición.....	80
3.5.1. Caso de Usos Reales.....	80
3.5.2. Interfaz y explicación de los casos de uso.....	81
3.6. Métricas de Calidad.....	91
3.6.1. Calidad del SODATD.....	91
3.6.2. Primera prueba.....	91
3.6.2.1. Funcionalidad.....	91
3.6.2.2. Fiabilidad.....	94
3.6.2.3. Usabilidad.....	94
3.6.2.4. Eficiencia.....	95
3.6.2.5. Mantenibilidad.....	95
3.6.2.6. Portabilidad.....	96
3.6.2.7. Confiabilidad.....	96
3.6.3. Segunda Prueba.....	97
3.6.3.1. Funcionalidad.....	97
3.6.3.2. Fiabilidad.....	99
3.6.3.3. Usabilidad.....	100
3.6.3.4. Eficiencia.....	100
3.6.3.5. Mantenibilidad.....	101
3.6.3.6. Portabilidad.....	101
3.6.3.7. Confiabilidad.....	101

3.7.	Tabla Comparativa de Pruebas.....	102
3.8.	Costos y Beneficios.....	103
3.8.3.	Análisis de costo.....	103
3.8.4.	Beneficios.....	106

CAPITULO IV

4.	Conclusiones.....	109
4.6.	Recomendaciones.....	110
4.7.	Bibliografía.....	111

ANEXO.



INDICE DE TABLAS

Tabla 2.1: Clasificación de las métricas de software.....	28
Tabla 2.2: Calculo de Punto Función.....	30
Tabla 2.3: Preguntas para el cálculo del factor de complejidad.....	32
Tabla 3.1: Requerimientos Funcionales.....	37
Tabla 3.2: Requerimientos no Funcionales.....	37
Tabla 3.3: Funciones del Sistema.....	38
Tabla 3.4: Descripciones de Actores.....	40
Tabla 3.5: Ingreso al sistema.....	42
Tabla 3.6: Registro de datos del Usuario.....	43
Tabla 3.7: Administración de Producto.....	45
Tabla 3.8: Administración de Cliente.....	46
Tabla 3.9: Elaborar Proforma.....	49
Tabla 3.10: Iniciar Sesión.....	82
Tabla 3.11: Lista de clientes.....	83
Tabla 3.12: Nuevo Cliente.....	84
Tabla 3.13: Elaboración Proforma.....	85
Tabla 3.14 Seleccionar producto.....	86
Tabla 3.15: nuevo producto.....	88
Tabla 3.16: registro de precio.....	89
Tabla 3.17: lista de usuario.....	91
Tabla 3.18.- Cálculo de puntos de función 1.....	92
Tabla 3.19.- Preguntas para el cálculo del factor de complejidad 1.....	93
Tabla 3.20.- Cálculo de puntos de función 2.....	97
Tabla 3.21.- Preguntas para el cálculo del factor de complejidad 2.....	98
Tabla 3.22.- Tabla Comparativa.....	103
Tabla 3.23.- Tabla de los Coeficientes para los Diferentes Proyectos.....	105
Tabla3.24. Tablas de los Costos del Sistema.....	106

INDICE DE FIGURAS

Figura 2.1: Cubo de Datos.....	13
Figura 2.2: Representación de tabla de hechos, dimensión y modelo estrella.....	15
Figura 2.3: Funcionamiento Mondrian.....	16
Figura 2.4: RUP.....	21
Figura 2.5: Diagrama de Caso de Uso.....	23
Figura 2.6: Diagrama de Clases.....	24
Figura 2.7: Diagrama Secuencia.....	25
Figura 2.8: Diagrama Estados.....	25
Figura 2.9: Diagrama Colaboración.....	26
Figura 2.10: Diagrama Despliegue.....	27
Figura 2.11: Parámetros.....	37
Figura 3.1: Diagrama de Casos de Uso Del Sistema.....	41
Figura 3.2: Diagrama de Casos de Uso Iniciar Sesión.....	42
Figura 3.3: Diagrama de Casos de Uso Administrar Producto.....	44
Figura 3.4: Diagrama de Casos de Uso Administrar Cliente.....	45
Figura 3.5: Diagrama de Casos de Uso Generar Proforma.....	47
Figura 3.6: Diagrama de Clases.....	50
Figura 3.7: Diagrama de Secuencia iniciar sesión.....	52
Figura 3.8: Diagrama de Secuencia Administrar Cliente.....	53
Figura 3.9: Diagrama de Secuencia Administrar Producto.....	54
Figura 3.10: Diagrama de Secuencia Elaborar Proforma.....	55
Figura 3.11: Diagrama de Secuencia Administrar Dimensión.....	56
Figura 3.12: Diagrama de Estados inicia sesión.....	57
Figura 3.13: Diagrama de Estados Administrar Cliente.....	58

Figura 3.14: Diagrama de Estados Administrar Producto.....	59
Figura 3.15: Diagrama de Estado Elaborar Proforma.....	60
Figura 3.16: Diagrama de Colaboración Inicias Sesión.....	61
Figura 3.17: Diagrama de Colaboración Administrar Cliente.....	61
Figura 3.18: Diagrama de Colaboración Administrar Producto.....	62
Figura 3.19: Diagrama de Colaboración Elaborar Proforma.....	62
Figura 3.20: Cubo (producto, medidas tiempo).....	63
Figura 3.21: Cubo (categoría producto, localización, tiempo).....	64
Figura 3.22: Diagrama de Componentes.....	65
Figura 3.23: Diagrama de Despliegue.....	66
Figura 3.24: Arquitectura del Sistema.....	67
Figura 3.25: Iniciar Sesión.....	68
Figura 3.26: Lista de Clientes.....	68
Figura 3.27: Elaboración de la proforma.....	69
Figura 3.28: Seleccionar Producto	69
Figura 3.29: Seleccionar Producto	70
Figura 3.30: Listado de Proforma.....	70
Figura 3.31: Registro de Venta	71
Figura 3.32: Listado de Ventas.....	71
Figura 3.33: Nuevo Producto	72
Figura 3.34: Lista de Productos.....	72
Figura 3.35: Nuevo Precio	73
Figura 3.36: Registro de Clientes	73
Figura 3.37: Registro de Oficinas Regionales.....	74
Figura 3.38: Vista Inicial.....	75
Figura 3.39: Vista desglosado el lugar	76
Figura 3.40: Vista con Grafico.....	77
Figura 3.41: Vista con Grafico.....	78
Figura 3.42: Vista con Grafico	79
Figura 3.43: Vista configurando dimensión	80
Figura 3.44: Iniciar Sesión.....	81

Figura 3.45: Lista de Clientes.....	82
Figura 3.46: Nuevo Cliente.....	83
Figura 3.47: Elaboración de Proforma	84
Figura 3.48: Seleccionar Producto:	85
Figura 3.49: Nuevo Producto.....	87
Figura 3.50: Lista de precios.....	88
Figura 3.51 Registro de precio.....	89
Figura 3.52: Lista de usuario.....	90



1. Introducción

En la actualidad los sistemas de información basados en computadoras se constituyen como herramientas principales para satisfacer las constantes necesidades en las organizaciones, lo cual permite adecuarse al avance tecnológico.

La arquitectura OLAP, que proviene de Online Analytical Processing (Procesamiento Analítico en Línea), define una tecnología que se basa en el análisis multidimensional de los datos y que le permite al usuario tener una visión más rápida e interactiva de los mismos.

Este análisis, también conocido como análisis del hipercono, organiza la información según los parámetros que se consulten, de tal manera que se realizara la estructura multidimensional y así se obtendrá la función requerida por la institución.

Esta estructura multidimensional es utilizada en el área de las ventas, informes, etc., especialmente porque la respuesta a sus consultas son complejas y se obtienen de manera rápida y precisa.

El Ministerio de Medio Ambiente y Agua, conlleva al incremento de información y junto a esto la complejidad en el procesamiento e interpretación de datos. Es en este contexto que se presentan técnicas para la exploración de datos como OLAP (OnLine Analytical Processing), que permite obtener información precisa de manera rápida y efectiva a través de cubos de decisión (denominados cubos OLAP) para la toma de decisiones.

El presente proyecto pretende desarrollar un Sistema de oferta y demanda para el apoyo a la toma de decisiones, tomando en cuenta la arquitectura OLAP orientada a servicios y aplicaciones web.

Se plantea, que el sistema de oferta y demanda para el apoyo a la toma de decisiones con OLAP, donde su labor principal será tener el control de la demanda de los clientes que realizan solicitudes de certificaciones¹, la oferta que se basa en una proforma², y así tener el control de las ventas realizadas. Entonces, a partir de esta información de manera rápida y sencilla se realizarán cálculos y diagramas estadísticos para poder realizar la toma de decisiones para el mejoramiento y manejo del SENAMHI.

1.1 Antecedentes

OLAP es el acrónimo en inglés de **procesamiento analítico en línea** (*On-Line Analytical Processing*). Es una solución utilizada en el campo de la llamada Inteligencia empresarial (o *Business Intelligence*) cuyo objetivo es agilizar la consulta de grandes cantidades de datos. Para ello utiliza estructuras multidimensionales (o Cubos OLAP) que contienen datos resumidos de grandes Bases de datos o Sistemas Transaccionales (OLTP). Se usa en informes de negocios de ventas, marketing, informes de dirección, minería de datos y áreas similares.

ROLAP Implementación OLAP que almacena los datos en un *motor relacional*. Los esquemas más comunes sobre los que se trabaja son *estrella* ó *copo de nieve*, aunque es posible trabajar sobre cualquier base de dato relacional.

La arquitectura está compuesta por un servidor de banco de datos relacional y el motor OLAP que se encuentra en un servidor. La principal ventaja de esta arquitectura es que permite el análisis de una gran cantidad de datos.

¹ Certificaciones: Certificaciones de datos, servicios y productos especializados

² Pro forma: Documento de cotización de venta de certificaciones.

Un **cubo OLAP**, es una base de datos multidimensional, en la cual el almacenamiento físico de los datos se realiza en un vector multidimensional.

1.1.1 Institucionales

El Servicio Nacional de Meteorología e Hidrología es una entidad estatal dependiente del Ministerio de Medio Ambiente y Agua, se encuentra ubicado en la calle Reyes Ortiz N° 41. Encargada de prestar Servicios de Meteorología e Hidrología al público en general en sus diferentes funciones teniendo así una amplia demanda de los clientes que solicitan certificaciones (certificaciones de datos, certificaciones de servicio y certificaciones de productos especializados) ofertados en el SENAMHI.

El SENAMHI cuenta con el funcionamiento de la oferta de certificaciones (datos, servicios y productos especializados) y Demanda de los Clientes (Instituciones Públicas, Universidades, Organizaciones no Gubernamentales y otros), que realizan solicitudes de certificaciones al SENAMHI, donde se evidencio que se llevan a cabo procesos semiautomáticos, los mismos que hacen que el trabajo avance con algunas falencias que se requiere mejorar.

Se cuenta con los siguientes tipos de certificaciones:

1. Certificación de datos Meteorológicos e Hidrológicos:
 - ✓ Datos Meteorológicos (precipitación, temperatura, humedad relativa, dirección prevalente, velocidad del viento, presión barométrica media, evaporación, horas sol, nubosidad, helada , granizo, nevada, etc.).
 - ✓ Datos Hidrológicos: (Niveles, caudales y aforos sólidos).
2. Certificación de servicios: (imágenes geográficas, otros).
3. Certificación de productos especializados:(fenómenos, otros).

Se cuenta con un registro de ventas de las certificaciones, donde se tienen los datos de las certificaciones que se realizan, entonces a partir de estos registros se debe realizar un informe que contenga la cantidad de ventas que se han realizado en cada oficina regional. Esta información esta almacenada en libros y documentos Excel así como también la oferta en el SENAMHI cuenta con un conjunto diferenciados de servicios y la manera de hacer conocer los mismos es mediante proformas. Se tiene dos tipos de proformas:

- ✓ Proforma con costo (información con un determinado precio)
- ✓ Proformas sin costo (información gratuita).

Las proformas son realizadas en base a parámetros, estaciones y los datos con diferentes escalas de tiempo real: anuales, mensuales, diarios ó de un determinado periodo.

Por otra parte se tiene el registro de las facturas y los depósitos bancarios de las ventas de certificaciones realizadas en libros Excel, clasificadas por fechas, para posteriormente realizar un informe, de las ventas que ha realizado el SENAMHI en cada Oficina Regional, para así obtener un informe general.

1.1.2 Otros Proyectos

FrameWork Spatial OLAP para el Soporte de la Toma de Decisiones, autor Miguel Angel Vega Pabon, Gestión 2009. Desarrolla un framework para el soporte a la toma de decisiones en base a soluciones propias.

Minería de datos en el sistema de información educativa, autor *Rojas Fernández, Glizeth, Gestión 2009* Aplica Minería de Datos sobre la base de datos del Ministerio de Educación y Culturas descubriendo características de la población estudiantil de las Unidades Educativas del área rural y urbana para conocer mejor el sector educativo y apoyar a la toma de decisiones.

Aplicación estocástica al sistema de inventarios de Vanguard Ltda. para la toma de decisiones Autor: Machaca Honorio, Ramiro, Gestión 2009. Desarrolla una aplicación estocástica al Sistema de inventario de Vanguard Ltda.. para la toma de decisiones que permita mejorar y agilizar los procesos de comercialización de productos y el control de estos en la empresa.

A diferencia de los proyectos mencionados el actual proyecto pretende desarrollar un sistema de oferta y demanda para el apoyo a la toma de decisiones, para un mejor funcionamiento de la institución.

1.2 Planteamiento del Problema

El análisis de la situación actual, permite observar diversos problemas que afectan el óptimo funcionamiento del Servicio Nacional de Meteorología e Hidrología SENAMHI dependiente del Ministerio de Medio Ambiente y Agua de los que se identificaron las siguientes causas:

- ✓ No se cuenta con el control digital de los registros de las demandas de los clientes que realizan solicitudes de certificaciones.
- ✓ No se cuenta con una base de datos de los registro de las ofertas realizadas en base a las proformas que fueron realizadas en la institución.
- ✓ No se cuenta con un recurso capaz de realizar el proceso estadístico y proporcionar la información lista para el análisis de oferta y demanda del SENAMHI.
- ✓ No se cuenta con un control centralizado y único de administración de todos estos procesos, lo cual conlleva a un impreciso y no oportuno proceso de reportes y análisis de ventas.
- ✓ No se cuenta con el registro de las facturas y depósitos bancarios de los ventas realizadas, para así realizar la clasificación por Oficinas Regionales.
- ✓ Se tiene un proceso lento y no confiable al realizar el control de la demanda y oferta para así realizar la planificación y las decisiones para el mejoramiento del SENAMHI.

Por todo lo anterior expuesto se propone el siguiente planteamiento del problema:

¿Cómo mejorar la planificación y la toma de decisiones respecto a la oferta de las certificaciones y la demanda de los clientes ya que está realizada en base a informaciones no reales?

1.3 Justificación.

1.3.1. Justificación Tecnológica

El desarrollo e implementación del sistema se realizará con software libre, la institución cuenta con equipos disponibles y recursos necesarios para la implementación del sistema y la eliminación de procesos manuales que originan demoras e inconsistencias de la información.

La necesidad de contar con una información a partir del análisis OLAP en cualquier momento es indispensable, por tanto se debe aprovechar la mayor cantidad de posibilidades para contar con esta información y capacidad de análisis.

1.3.2 Justificación Económica

Cada Institución pública tiene la necesidad de reducir los costos y de tener un máximo rendimiento. Es de esta manera que el sistema a desarrollar no tiene un costo monetario ya que se cuenta con el hardware necesario y el software de carácter gratuito y libre.

Entonces se sistematizarán las tareas que se realizan manualmente o semiautomáticamente en la institución, con la implementación del sistema en el proceso de la toma de decisiones se realizará, un mejor funcionamiento en la institución de manera rápida, confiable y precisa.

1.3.3 Justificación Social

El sistema facilitará los cálculos estadísticos, apoyará y proporcionará al sector administrativo (oficinas regionales) de la institución será una herramienta que colabore y facilite las actividades diarias, especialmente para el análisis y apoyo a las decisiones que la institución realiza, y así obtener un mejor funcionamiento y atención hacia el cliente.

El sistema permitirá ahorrar tiempo en el proceso de elaboración de diagramas estadísticos y así aumentar las ganancias de la institución.

1.4 Objetivos

1.4.1 Objetivo General

Desarrollar un sistema de apoyo a la toma de decisiones sobre la oferta de certificaciones y la demanda de los clientes. En base a datos confiables, oportunos y reales, a través de herramientas orientadas a la toma de decisiones OLAP (*OnLine Analytical Processing* o procesamiento Analítico En Línea) y software libre, para el SENAMHI.

1.4.2 Objetivo Especifico

- ✓ Se realizará el diseño de la base datos, que almacene toda la información, de la oferta y demanda de las ventas de certificaciones.
- ✓ Utilizar la estructura multidimensional (o **Cubos OLAP**) que contienen datos resumidos de una Bases de datos Relacional.
- ✓ Aplicará la arquitectura OLAP para obtener resultados de forma más eficiente y rápida.
- ✓ Métodos funciones para la visualización de diagramas estadísticos.

Desarrollar el producto bajo una arquitectura orientada a aplicaciones WEB y arquitectura orientada a objetos.

1.5 Alcances

Luego de analizar los problemas identificados, el presente proyecto desarrolla un Sistema de Oferta y Demanda para el Apoyo a la Toma de Decisiones con OLAP conformada por los siguientes módulos:

Modulo oferta y demanda

- ✓ Se tiene una estructura multidimensional (o Cubos OLAP) en la elaboración de la base de datos y así ofrecer una velocidad de las consultas requeridas por la institución.

Modulo centralización de la información de la oferta y demanda

- ✓ Realiza el seguimiento de las ventas de certificaciones que se realizan en la institución.
- ✓ El control de la demanda y oferta de la institución en base a las ventas realizadas en la institución.
- ✓ El control de los registros de las facturas y depósitos bancarios.

Aspectos que no competen al proyecto

- ✓ No realizar el Registro de los datos Meteorológicos e Hidrológicos obtenidos en la institución.
- ✓ No realizara reportes contables
- ✓ No realizar el Registro de Imágenes Satelitales o geográficas de la institución.

1.6 Herramientas

Está compuesto con servidor de base de datos POSTGRES para el proceso de datos de información en línea, un servidor de aplicaciones Java 2 Enterprise Edition para el alojamiento del sistema de prueba y servicio WEB, el servidor Mondrian para la aplicación del Cubo OLAP.



2. MARCO TEORICO

2.1 OLAP (Online Analytical Processing)

OLAP procesamiento analítico en línea (*On-Line Analytical Processing*). Es una solución utilizada en el campo de la Inteligencia de Negocios (Business Intelligence) cuyo objetivo es agilizar la consulta de grandes cantidades de datos. Para ello utiliza estructuras multidimensionales (o Cubos OLAP) que contienen datos resumidos de grandes Bases de Datos o Sistemas Transaccionales (OLTP). Se usan en informes de negocios de ventas, marketing, informes de dirección, minería de datos y áreas similares.

La razón de usar OLAP para las consultas es la velocidad de respuesta. Una base de datos relacional almacena entidades en tablas discretas si han sido normalizadas. Esta estructura es buena en un sistema OLTP pero para las complejas consultas multitabla es relativamente lenta. Un modelo mejor para búsquedas es una base de datos multidimensional.

En la base de cualquier sistema OLAP se encuentra el concepto de cubo OLAP (también llamado cubo multidimensional o hiper cubo). Se compone de hechos numéricos llamados **medidas** que se clasifican por dimensiones. El cubo de metadatos es típicamente creado a partir de un esquema en estrella o copo de nieve, esquema de las tablas en una base de datos relacional. Las medidas se obtienen de los registros de una tabla de hechos y las dimensiones se derivan de la dimensión de los cuadros.

OLAP o Proceso analítico en línea es el nombre formal para el análisis de cubos multidimensionales - una forma más intuitiva de ver la información empresarial. Con OLAP se puede ver un conjunto de datos de su negocio de muchas y

diversas formas sin mucho esfuerzo. Los archivos OLAP o cubos modelan los datos en **dimensiones**. Una dimensión es una clasificación de alguna actividad en una organización por la cual se puede medir su éxito. Por ejemplo, puede monitorear sus ventas contra los productos o clientes en un periodo de tiempo.

Dimensiones regulares son aquellos datos que se quieren medir, por ejemplo, si desea seguir el control de sus ventas, puede utilizar:

Clientes: Quienes son los mejores, donde se encuentran, que es lo que compran?

Productos: Con respecto a los clientes, quien los compra? Que productos se están vendiendo?

Tiempo: Como voy ahora con respecto al último año o último mes?

En otro tipo de aplicaciones, por ejemplo cuentas por cobrar, se pueden utilizar dimensiones como el Tiempo para llevar control del vencimiento de sus documentos. En contabilidad, una dimensión podría ser su catalogo de cuentas, etc.

Estas dimensiones se conforman de elementos que están dispuestos en niveles jerárquicos o simplemente niveles. Los niveles pueden ser por ejemplo, país, estado, ciudad. Se puede navegar a través de esta jerarquía a través de los niveles o a través de sus elementos. [PENTAHO, 2007]

Hay dos operaciones básicas que se pueden realizar en un cubo OLAP:

Rotar y Rebanar: Usted puede cambiar las dimensiones del cubo que está viendo y obtener una nueva vista de información. Por ejemplo, 'Ventas por producto' puede cambiarse fácilmente a 'Ventas por vendedor'. Rebanar es cambiar el valor de una dimensión por otro valor, por ejemplo, de las ventas de Enero a las ventas de Febrero. Rotar es aventar el cubo como si fuera un dado para obtener una nueva cara del cubo.

Con esta simple combinación de cosas, se puede abrir la información generada por un negocio o información corporativa para todo el personal tomador de decisiones en formas que antes no era posible realizarlo.

2.2 Cubo OLAP

Los cubos son elementos claves en OLAP (online analytic processing), una tecnología que provee rápido acceso a datos en un almacén de datos (data warehouse). Los cubos proveen un mecanismo para buscar datos con rapidez y tiempo en respuesta uniforme independientemente de la realizada de la cantidad de datos en el cubo o la complejidad del procedimiento de búsqueda.

Los cubos son subconjuntos de datos de un almacén de datos, organizado y sumariado dentro de una estructura multidimensional. Los datos se sumarizan de acuerdo a factores de negocio seleccionados, proveyendo el mecanismo para la rápida y uniforme tiempo de respuesta de las complejas consultas.

Un cubo OLAP es una estructura de datos que permite un rápido análisis de datos. El definir los datos dentro de un cubo supera la limitación de tiempo de respuesta que tienen las bases de datos relacionales. Las bases de datos relacionales no están bien adecuadas para un análisis instantáneo y visualización de una gran cantidad de datos. En vez de ello, están bien adecuadas para crear registros a partir de una serie de transacciones particulares. **[MICROSOFT, 2000]**

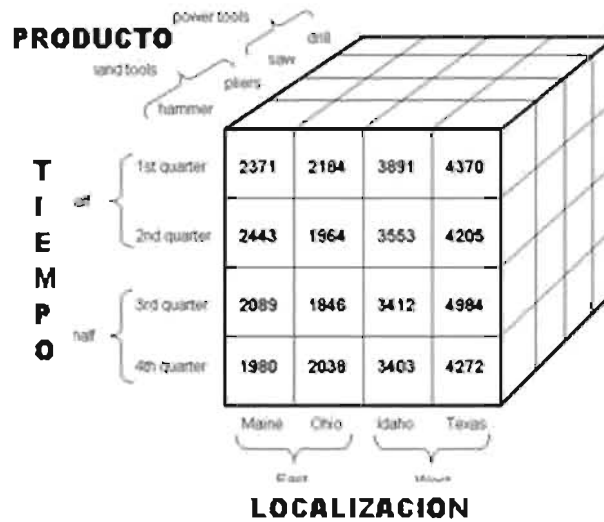


Figura 2.1: Cubo de Datos
Fuente: [Elaboración Propia, 2010]

2.3 Data Warehouse

Un **Data warehouse** es una base de datos corporativa que se caracteriza por integrar y depurar información de una o más fuentes distintas, para luego procesarla permitiendo su análisis desde infinidad de perspectivas y con grandes velocidades de respuesta. La creación de un datawarehouse representa en la mayoría de las ocasiones el primer paso, desde el punto de vista técnico, para implantar una solución completa y fiable de Business Intelligence.

La ventaja principal de este tipo de bases de datos radica en las estructuras en las que se almacena la información (modelos de tablas en estrella, en copo de nieve, cubos relacionales... etc). Este tipo de persistencia de la información es homogénea y fiable, y permite la consulta y el tratamiento jerarquizado de la misma (siempre en un entorno diferente a los sistemas operacionales). [INMON, 2010]

2.4 Modelo Estrella

Es el estilo más simple de un **esquema data warehouse**. El esquema estrella está compuesto de algunas tablas de hechos (*posiblemente una sola, justificando el nombre*) que referencian algún número de tablas dimensión. El esquema estrella es considerado un importante caso especial del esquema de copo de nieve (snowflake). [Wikipedia, 2008]

2.5 Tabla de Hechos

En las bases de datos, y más concretamente en un data warehouse, una **tabla de hechos** (o *tabla fact*) es la tabla central de un esquema dimensional (en estrella o en copo de nieve) y contiene los valores de las medidas de negocio. Cada medida se toma mediante la intersección de las dimensiones que la definen, dichas dimensiones estarán reflejadas en sus correspondientes tablas de dimensiones que rodearán la tabla de hechos y estarán relacionadas con ella.

En la tabla central (**Ventas**) es la tabla de hechos de un diseño de modelo de datos en estrella, las cinco tablas que la rodean (**Ubicacion, Tiempo, Producto y Cliente**) son las cuatro dimensiones de que consta esta tabla de hechos, en dicha tabla se almacenan, en este caso, las **unidades** vendidas y el **precio** obtenido por dichas ventas, estos son los **hechos** o medidas de negocio almacenados y que, gracias al diseño multidimensional en estrella, podrán ser analizados de forma exhaustiva, típicamente mediante técnicas OLAP (*procesamiento analítico on-line*). [Wikipedia, 2010],

2.6 Tabla Dimensión

Una dimensión es un elemento dato que categoriza cada ítem en un conjunto de datos dentro de regiones que no se superponen. Provee el significado de “*slice and dice*” datos en un data warehouse. La funcionalidad primordial de las

dimensiones es triple: proveer filtrado, agrupación y etiquetado. Por ejemplo en un data warehouse cada persona es categorizada de acuerdo a un género de masculino, femenino o desconocido, un usuario de este data warehouse sería capaz entonces de filtrar o categorizar cada presentación o reporte tan solo por filtrado de la dimensión de género, o mostrar resultados generados por género.

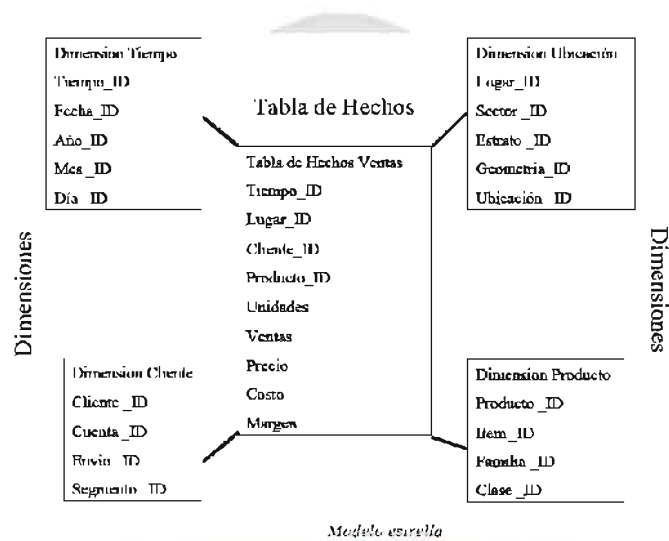


Figura 2.2: Representación de tabla de hechos, dimensión y modelo estrella
Fuente: [Elaboración Propia, 2010]

2.7 Mondrian

Mondrian es una de las aplicaciones más importantes de la plataforma Pentaho BI. Mondrian es un servidor OLAP open source que gestiona comunicación entre una aplicación OLAP (escrita en Java) y la base de datos con los datos fuente. Es decir, Mondrian actúa como “JDBC para OLAP”. [PENTAHO, 2007]

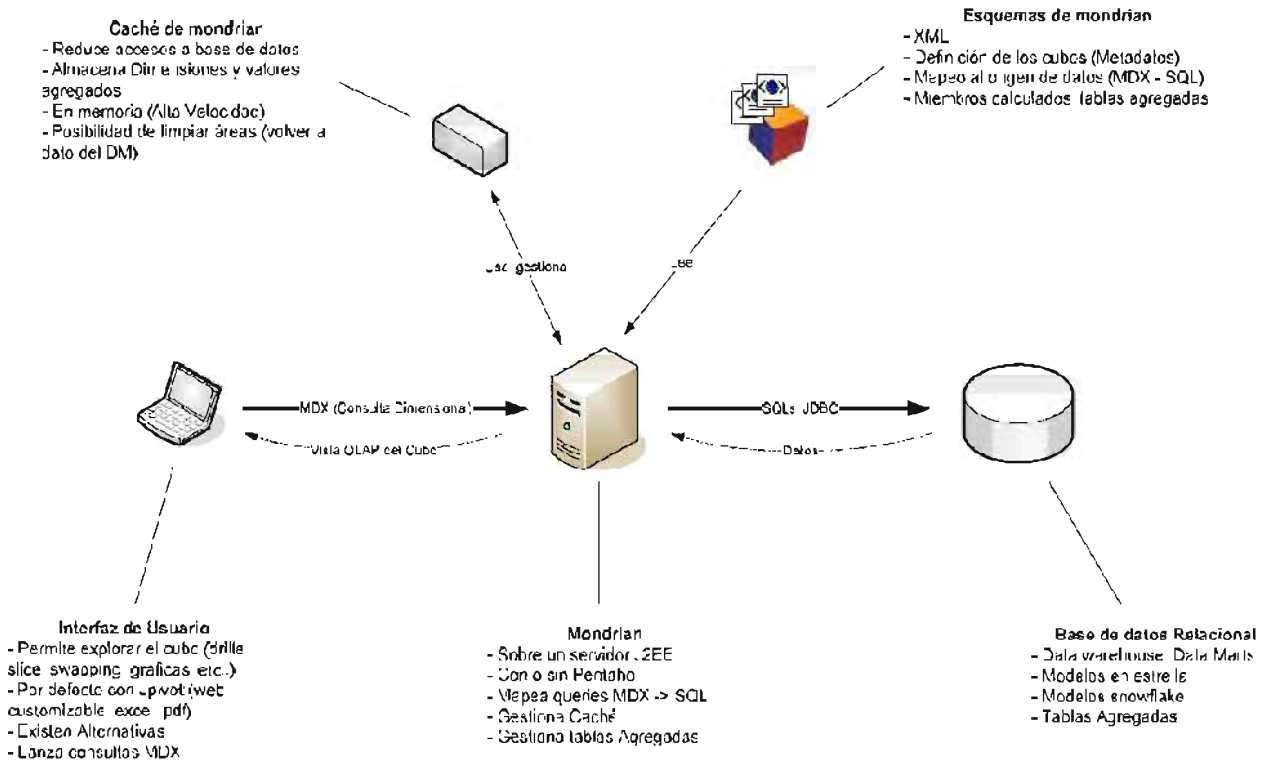


Figura 2.3: Funcionamiento Mondrian
Fuente: [Elaboración Propia, 2010]

2.8 JPivot

JPivot es una librería de componentes JSP que se utiliza para construir tablas OLAP generadas de forma dinámica. Este tipo de tablas es de gran utilidad ya que permite mostrar los resultados de las consultas filtrando por los campos de la tabla de manera que se puedan quitar y poner distintos criterios de búsqueda de los datos, consiguiendo un amplio abanico de posibilidades.

Para desarrollar este tipo de tablas pivotantes es necesaria la arquitectura OLAP Mondrian contenida en la plataforma Pentaho. Esta arquitectura que corre sobre un servidor web nos permite la comunicación entre aplicaciones OLAP con bases de datos. El núcleo del servidor Mondrian es similar a JDBC pero exclusivo para

OLAP. Proporciona la conexión a la base de datos y ejecuta las sentencias SQL. [PENTAHO, 2007]

2.9 Java 2 Platform, Enterprise Edition (J2EE)

Java Platform, Enterprise Edition o Java EE (anteriormente conocido como Java 2 Platform, Enterprise Edition o J2EE hasta la versión 1.4), es una plataforma de programación que parte de la Plataforma Java para desarrollar y ejecutar software de aplicaciones en Lenguaje de programación Java con arquitectura de N niveles distribuida, basándose ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones. La plataforma Java EE está definida por una *especificación*. Similar a otras especificaciones del Java Community Process, Java EE es también considerada informalmente como un estándar debido a que los suministradores deben cumplir ciertos requisitos de conformidad para declarar que sus productos son *conformes a Java EE*; estandarizado por The Java Community Process / JCP.

Java EE incluye varias especificaciones de API, tales como JDBC, RMI, e-mail, JMS, Servicios Web, XML, etc. y define cómo coordinarlos. Java EE también configura algunas especificaciones únicas para Java EE para componentes. Estas incluyen Enterprise JavaBeans, servlets, portlets (siguiendo la especificación de Portlets Java), JavaServer Pages y varias tecnologías de servicios web. Esto permite al desarrollador crear una Aplicación de Empresa portable entre plataformas y escalable, a la vez que integrable con tecnologías anteriores. Otros beneficios añadidos son, por ejemplo, que el servidor de aplicaciones puede manejar transacciones, la seguridad, escalabilidad, concurrencia y gestión de los componentes desplegados, significando que los desarrolladores pueden concentrarse más en la lógica de negocio de los componentes en lugar de en tareas de mantenimiento de bajo nivel. [CRAWFORD, 2003]

2.10 Java Server Faces

JavaServer Faces (JSF) es una tecnología y framework para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE. **JSF** usa JavaServer Pages (JSP) como la tecnología que permite hacer el despliegue de las páginas, pero también se puede acomodar a otras tecnologías como XUL.

JSF incluye:

- ✓ Un conjunto de APIs para representar componentes de una interfaz de usuario y administrar su estado, manejar eventos, validar entrada, definir un esquema de navegación de las páginas y dar soporte para internacionalización y accesibilidad.
- ✓ Un conjunto por defecto de componentes para la interfaz de usuario.
- ✓ Dos bibliotecas de etiquetas personalizadas para JavaServer Pages que permiten expresar una interfaz JavaServer Faces dentro de una página JSP.
- ✓ Un modelo de eventos en el lado del servidor.
- ✓ Administración de estados.
- ✓ Beans administrados.

Estos objetivos de diseño representan el foco de desarrollo de JSF:

- ✓ Definir un conjunto simple de clases base de Java para componentes de la interfaz de usuario, estado de los componentes y eventos de entrada. Estas clases tratarán los aspectos del ciclo de vida de la interfaz de usuario, controlando el estado de un componente durante el ciclo de vida de su página.
- ✓ Proporcionar un conjunto de componentes para la interfaz de usuario, incluyendo los elementos estándares de HTML para representar un formulario. Estos componentes se obtendrán de un conjunto básico de clases base que se pueden utilizar para definir componentes nuevos.

- ✓ Proporcionar un modelo de JavaBeans para enviar eventos desde los controles de la interfaz de usuario del lado del cliente a la aplicación del servidor
- ✓ Definir APIs para la validación de entrada, incluyendo soporte para la validación en el lado del cliente.
- ✓ Especificar un modelo para la internacionalización y localización de la interfaz de usuario.
- ✓ Automatizar la generación de salidas apropiadas para el objetivo del cliente, teniendo en cuenta todos los datos de configuración disponibles del cliente, como versión del navegador. [CAY,2007]

2.11 Ajax

Ajax, acrónimo de *Asynchronous JavaScript And XML* (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones. Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se requieren al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página.

JavaScript es el lenguaje interpretado (scripting language) en el que normalmente se efectúan las funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante *XMLHttpRequest*, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML. Ajax es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores dado que está basado en estándares abiertos como JavaScript y Document Object Model (DOM).

2. 12 PostgreSQL

es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales.

PostgreSQL utiliza un modelo cliente/servidor y usa *multiprocesos* en vez de *multihilos* para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando. [PostgreSQL, 2010]

2. 13 Lenguaje de modelo Unificado UML

Lenguaje Unificado de Modelado (LUM o UML, por sus siglas en inglés, *Unified Modeling Language*) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

Es importante resaltar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos.

Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo. [EVITTS , 2000]

2. 14 Proceso Unificado de Desarrollo de Software (RUP)

La metodología RUP, llamada así por sus siglas en inglés Rational Unified Process, divide en 4 fases el desarrollo del software:

- ✓ **Inicio**, El Objetivo en esta etapa es determinar la visión del proyecto.
- ✓ **Elaboración**, En esta etapa el objetivo es determinar la arquitectura óptima.
- ✓ **Construcción**, En esta etapa el objetivo es llevar a obtener la capacidad operacional inicial.
- ✓ **Transición**, El objetivo es llegar a obtener el reléase del proyecto.

Cada una de estas etapas es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los Objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes. Vale mencionar que el ciclo de vida que se desarrolla por cada iteración, es llevada bajo dos disciplinas [JACOBSON, BOOCH, RUMBAUGH, 2000.]

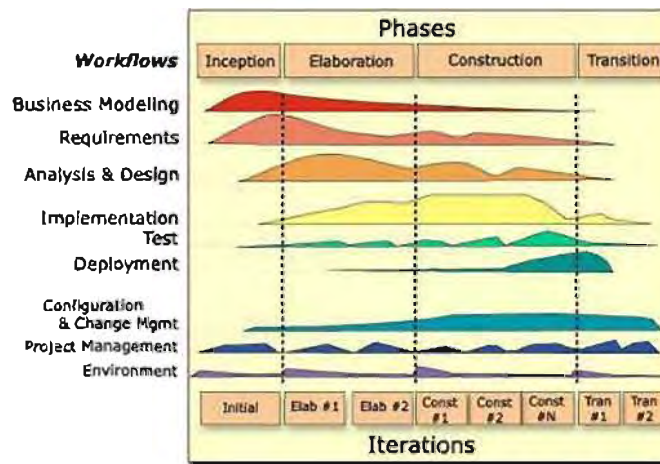


Figura 2.4: RUP

Fuente: [Consolidat, 2008]

2. 15 Las cuatro fases de desarrollo del RUP

FASE INICIO

Antes de iniciar un proyecto es conveniente planearse algunas cuestiones ¿Cuál es el objetivo? ¿Es factible? ¿Lo construimos o lo comparamos? ¿Cuánto va a costar? La fase de inicio trata de responder a estas preguntas y a otras más. Sin embargo no pretendemos una estimación precisa o la captura de todos los requisitos. Más bien se trata de explorar el problema lo justo para decidir si vamos a construir o dejarlo. Generalmente no debe durar mucho más de una semana.

Los objetivos son:

- ✓ Establecer el ámbito del proyecto y sus límites.
- ✓ Encontrar los casos de uso críticos del sistema, los escenarios básicos que definen la profundidad.
- ✓ Mostrar al menos una arquitectura candidata para los escenarios principales.
- ✓ Estimar el coste en recursos y tiempo de todo el proyecto.
- ✓ Estimar los riesgos, las fuentes de incertidumbre.

Los productos de la fase de inicio deben ser:

- ✓ Visión del negocio: Describe los objetivos y restricciones a alto nivel
- ✓ Modelo de casos de uso.

FASE DE ELABORACION

El propósito de la fase de elaboración es analizar el dominio del problema, establecer los cimientos de la arquitectura, desarrollar el plan del proyecto y eliminar los mayores riesgos.

En esta fase se construye los prototipos de la arquitectura, que debe evolucionar en iteraciones sucesivas hasta convertirse en el sistema final. Este prototipo debe contener los casos de uso críticos identificados en la fase de inicio. La forma de aproximarse a esta fase debe ser tratar de abarcar todo el proyecto con la portabilidad mínima. Solo se profundiza en los puntos críticos de la arquitectura o Riesgos importantes. [JACOBSON, BOOCH, RUMBAUGH, 2000.]

DIAGRAMA DE CASOS DE USO

Sirve para describir las interacciones del sistema con su entorno, identificando los **Actores**, que representan los diferentes roles desempeñados por los usuarios del sistema, y los **Casos de Uso**, que corresponden a la funcionalidad que el sistema ofrece a sus usuarios, explicada desde el punto de vista de éstos. Los actores no son solamente humanos, pueden ser también otros sistemas con los cuales el sistema en desarrollo interactúa de alguna manera. [JACOBSON, BOOCH, RUMBAUGH, 2000.]

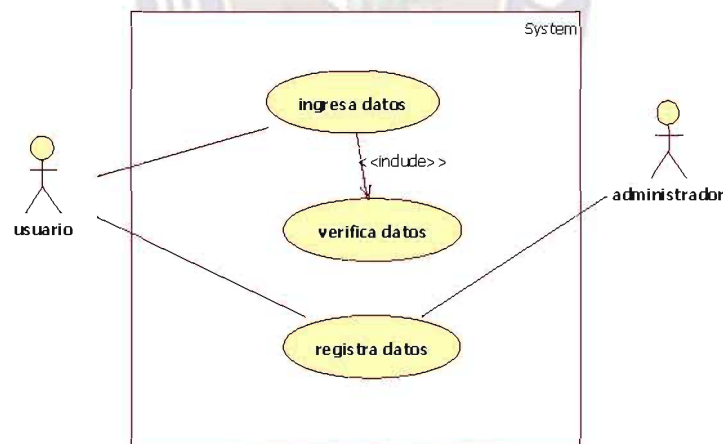


Figura 2.5: Diagrama de Caso de Uso

Fuente: [Elaboración Propia 2010]

DIAGRAMA DE CLASES

Un **diagrama de clases** es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los

sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro. [JACOBSON, BOOCH, RUMBAUGH, 2000.]

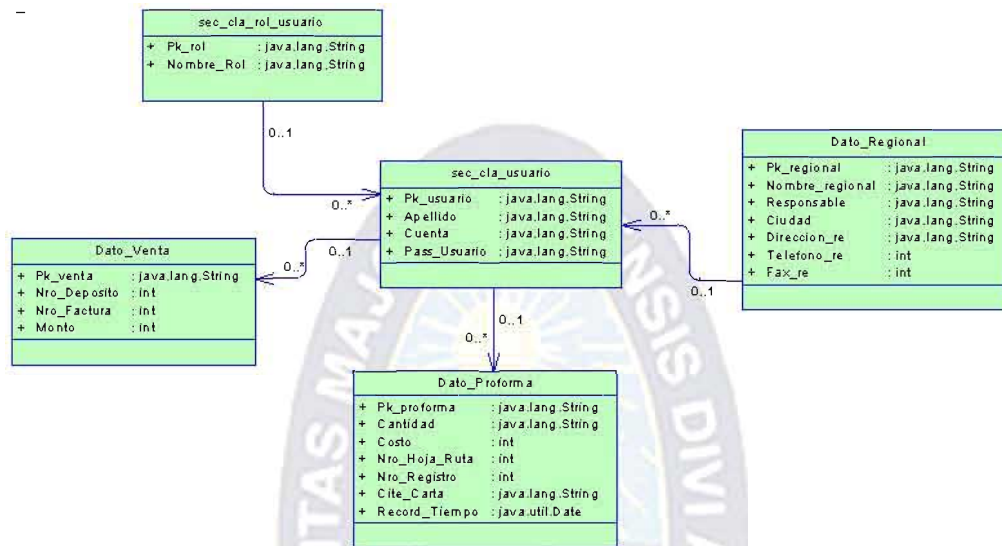


Figura 2.6: Diagrama de Clases
Fuente: [Elaboración Propia, 2010]

DIAGRAMA DE SECUENCIAS

Un **diagrama de secuencia** muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso. Mientras que el diagrama de casos de uso permite el modelado de una vista *business* del escenario, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y mensajes intercambiados entre los objetos. [JACOBSON, BOOCH, RUMBAUGH, 2000.]

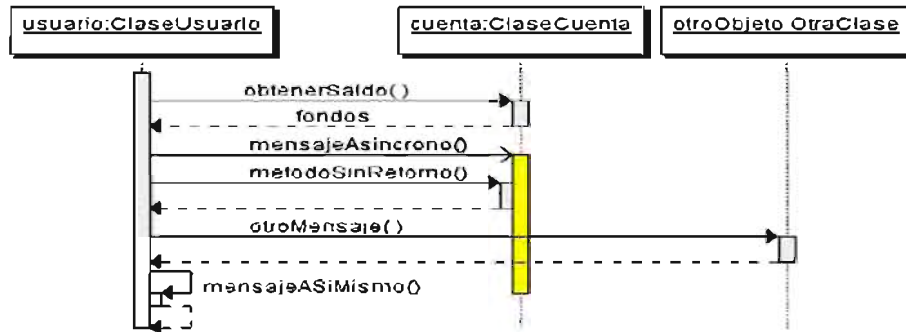


Figura 2.7: Diagrama Secuencia
Fuente: [wikipedia, 2010]

DIAGRAMA DE ESTADOS

Representan la secuencia de estados por los que un objeto o una interacción entre objetos para durante su tiempo de vida en respuesta a estímulos (eventos) recibidos. Representa lo que podemos denominar en conjunto una maquina de estados. Un estado en UML es cuando un objeto o una interacción satisfacen una condición, desarrollo o alguna acción o se encuentra esperando un evento. [JACOBSON, BOOCH, RUMBAUGH, 2000.]

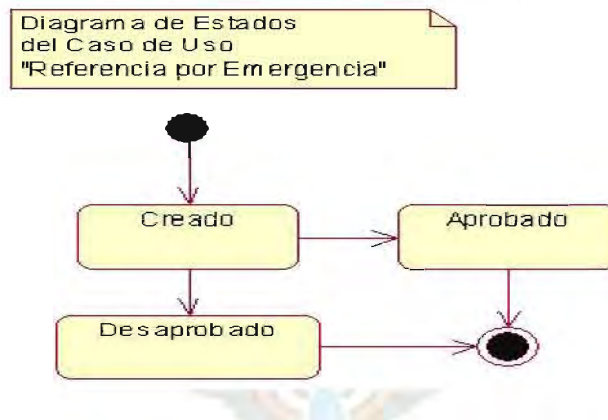


Figura 2.8: Diagrama estados
Fuente: [Wikipedia, 2010]

DIAGRAMA DE COLABORACION

Un Diagrama de Colaboración es esencialmente un diagrama que muestra interacciones organizadas alrededor de los roles. A diferencia de los diagramas de

secuencia, los diagramas de comunicación muestran explícitamente las relaciones de los roles. [JACOBSON, BOOCH, RUMBAUGH, 2000.]

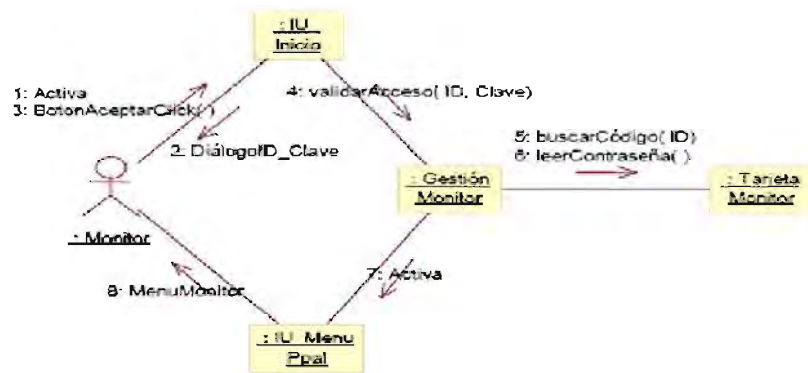


Figura 2.9: Diagrama Colaboración
Fuente: [Wikipedia, 2010]

FASE DE CONSTRUCCION

La finalidad principal de esta fase es alcanzar la capacidad operacional del producto de forma incremental a través de las sucesivas iteraciones. Durante esta fase todos los componentes, características y requisitos, que no lo hayan hecho hasta ahora, han de ser implementados, integrados y probados, obteniéndose una versión del producto que se pueda poner en manos de los usuarios. El énfasis en esta fase se pone a controlar las operaciones realizadas, administrándolos recursos eficientemente, de tal forma que se optimicen los coste de calendario y de calidad.

DIAGRAMA DE COMPONENTES

Un diagrama de componentes muestra la organización y las dependencias entre un conjunto de componentes. Los diagramas de componentes cubren la vista de implementación estática de un sistema. Se relacionan con los diagramas de clases en que un componente se corresponde, por lo común, con una o más clases, interfaces o colaboraciones. [JACOBSON, BOOCH, RUMBAUGH, 2000].

DIAGRAMA DE DESPLIEGUE

Los diagramas de despliegue muestran la estructura del sistema en tiempo de ejecución. En un diagrama de este tipo se puede conocer como se configuran e implementan los elementos de hardware y software que forman una aplicación. Los diagramas de despliegue constan de nodos, componentes, y las relaciones entre ellos. [JACOBSON, BOOCH, RUMBAUGH, 2000]

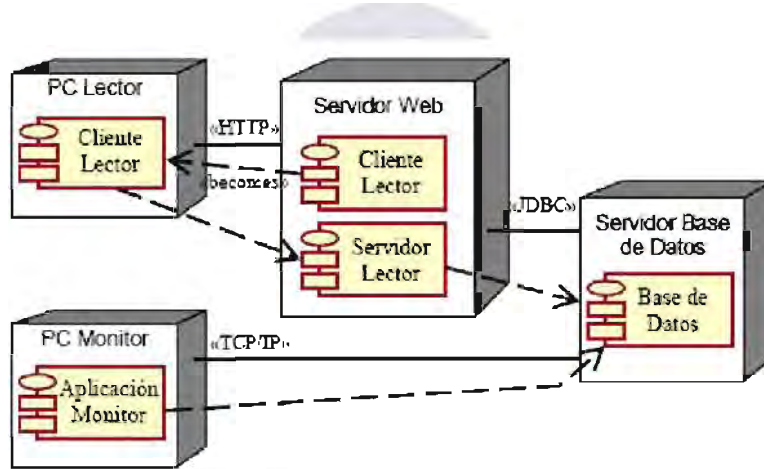


Figura 2.10: Diagrama Despliegue

Fuente: [Wikipedia, 2010]

FASE DE TRANSICION

La finalidad de la fase de transición es poner el producto en mano de los usuarios finales, para lo que típicamente se requiere desarrollar nuevas versiones actualizadas del producto, completar la documentación, entrenar al usuario en el manejo del producto y en general, tareas relacionadas con el ajuste, configuración, instalación y usabilidad del producto. [JACOBSON, BOOCH, RUMBAUGH, 2000.]

2. 16 Métricas De Calidad De Software

Todas las metodologías y herramientas tienen un único fin producir software de gran calidad.

El concepto de métrica es el término que describe muchos y muy variados casos de medición. Siendo una métrica una medida estadística (no cuantitativa como en otras disciplinas ejemplo física) que se aplica a todos los aspectos de calidad de

software, los cuales deben ser medidos desde diferentes puntos de vista como el análisis, construcción, funcional, documentación, métodos, proceso, usuario, entre otros.

Seguidamente se presentan algunas definiciones de Calidad de Software según algunos autores: “Concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente”. [PRESS 2002].

2.17. Clasificación De Las Métricas De Software

De complejidad	Métricas que definen la medición de la complejidad: volumen, tamaño, anidaciones, y configuración.
De calidad	Métricas que definen la calidad del software: exactitud, estructuración o modularidad, pruebas, mantenimiento.
De competencia	Métricas que intentan valorar o medir las actividades de productividad de los programadores con respecto a su, rapidez y competencia
De desempeño	Métricas que miden la conducta de módulos y sistemas de un software, bajo la supervisión del SO o hardware.
Estilizadas	Métricas de experimentación y de preferencia: estilo de código, convenciones, limitaciones, etc.

Tabla 2.1: Clasificación de las métricas de software

Fuente: [Press, 2002]

2.18. Métricas De Software Orientadas a la Función

Medida. Proporciona una indicación cuantitativa de extensión, cantidad, dimensiones, capacidad y tamaño de algunos atributos de un proceso o producto. Pueden ser directas, por ejemplo el número de líneas de código, número de errores encontrados, etc., o pueden ser indirectos por ejemplo la funcionalidad, calidad, complejidad, etc.

Medición. Se refiere al acto de terminar una medida.

Indicador. Es una métrica o combinación de métricas que proporcionan una visión del proceso, del proyecto o del software en sí, y poder hacer ajustes para que los procesos mejoren.

La medida de punto función se propuso en 1979 y trata de medir la funcionalidad o utilidad del software.

2.19 ISO 9126

International Organization for Standardization (ISO) / International Electric Comity (IEC). Este estándar define el empleo de métricas como una contribución relativamente independiente a la calidad del software, asociada con el diseño y la evaluación de la interfaz de usuario y su interacción con el usuario final.

La norma ISO/IEC 9126 tiene a su vez una clasificación importante, la misma plantea la utilización de seis métricas, en este sentido solamente se hizo énfasis en las métricas orientadas al contexto de uso, que en este caso son cuatro (funcionalidad, fiabilidad, usabilidad y eficiencia), a continuación se describen una a una incluyendo las llamadas Submétricas, que no son más que los aspectos que miden cada una de las métricas. [Kobayashi, 2007]

2.19.1. Funcionalidad

Cálculo Del Punto Función

Para el cálculo del punto función se deberá completar la siguiente tabla de valores del dominio de la información, por ejemplo: [PRESS 2002]

Parámetro	Cuenta	Factor de ponderación			Subtotal
		Simple	Medio	Complejo	
Número de entradas de usuario	X	3	4	6	=
Número de salidas de usuario	X	4	5	7	=
Número de peticiones de usuario	X	3	4	6	=
Número de archivos	X	7	10	15	=
Número de interfaces externas	X	5	7	10	=
Total					

Tabla 2.2: Calculo de Punto Función

Fuente: [Press, 2002]

Donde:

Entradas de usuario: Son entradas que proporcionan diferentes datos a la aplicación. No se debe confundir las entradas con las peticiones de usuario.

Salidas de usuario: Son reportes, pantallas o mensajes de error que proporcionan información. Los elementos de un reporte, no se cuentan de forma separada.

Peticiones de usuario: Es una entrada interactiva que produce la generación de alguna respuesta del software en forma de salida interactiva.

Archivos: Son los archivos que pueden ser parte de una base de datos o independientes.

Interfaces externas: Son los archivos que se usan para transmitir información a otro sistema.

Luego se debe responder a cada una de las siguientes catorce preguntas y asignarles un valor entre **0** y **5**.

Donde:

0= sin influencia, **1=** incidental, **2=** moderado, **3=** medio, **4=** significativo, **5=** esencial

No	PREGUNTAS SOBRE LAS CARACTERÍSTICAS DEL SISTEMA	VALORACIÓN
1	¿Requiere el sistema de copias de seguridad y de recuperación?	
2	¿Se requiere comunicación de datos?	
3	¿Existen funciones de procesamiento distribuido?	

4	¿El rendimiento es crítico?	
5	¿Será ejecutado el sistema en un entorno operativo existente y fuertemente utilizado?	
6	¿Se requiere entrada de datos en línea?	
7	¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas o variadas opciones?	
8	¿Se actualizan los datos maestros de forma interactiva?	
9	¿Son complejas las entradas, las salidas, los archivos o las peticiones?	
10	¿Es complejo el procesamiento interno?	
11	¿Se ha diseñado el código reutilizable?	
12	¿Están incluidas en el diseño la conversión y la instalación?	
13	¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	
14	¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?	
FACTOR DE COMPLEJIDAD		$\sum F_i =$

Tabla 2.3: Preguntas para el cálculo del factor de complejidad

Fuente: [Press, 2002]

Sumar los puntos asignados a cada respuesta y obtener un total **F** que indica un valor de ajuste de complejidad.

Para calcular el punto función se utiliza la siguiente relación:

$$PF = CuentaTotal * (X + Min(Y)) * \sum F_i$$

Donde:

PF: Medida de funcionalidad.

Cuenta Total: Es la suma de los parámetros del dominio de la información.

X: Confiabilidad del proyecto, varía entre 1% a 100%.

Min(Y): Error mínimo aceptable al de la complejidad.

$\sum F_i$: Son los valores de ajuste de complejidad, donde $(1 \leq i \leq 14)$.

Luego para calcular el punto Función (PF) se emplea la siguiente relación

$$PF = CuentaTotal * (X + Min(Y)) * \sum F_i$$

Con $X=0.65$ (Pie de corrección), $Min(Y)=0.01$ (Error de la confiabilidad del sistema), además de los valores de $\sum F_i$ y *CuentaTotal* obtenidos de las tablas.

Reemplazando:

$$PF = CuentaTotal * (X + Min(Y)) * \sum F_i$$

$$PF = CuentaTotal * (0,65 + 0.01 * \sum F_i)$$

Ahora realizamos la comparación de funcionalidad del sistema con el PF máximo que se puede alcanzar es decir que $\sum F_i = 70$.

$$PF = CuentaTotal * (0,65 + 0.01 * 70)$$

PF_{MAX}

Con máximos valores de ajuste a la complejidad se tiene que la funcionalidad real es:

$$FUNCIONALIDAD = \left(\frac{PF}{PF_{MAX}} \right) * 100\%$$

2.19.2. Fiabilidad

Para Obtener un valor numérico aproximado de la fiabilidad del prototipo debemos realizar los cálculos de “El tiempo medio entre fallas” y “La disponibilidad mostrada a continuación”

MTTF: Tiempo medio hasta la siguiente falla

MTTR: Tiempo medio de reparación

MTBF: Tiempo medio entre fallas

$$MTBF = MTTF + MTTR$$

Ahora se debe calcular la disposición usando la siguiente formula

$$FIABILIDAD = D = [MTTF / (MTTF + MTTR)]$$

2.19.3. Usabilidad

Test de usuario: El test de usuario consiste en realizar una evaluación escrita después de la realización de las pruebas en los usuarios finales.

2.19.4. Eficiencia

Dentro de los parámetros que se debe medir, se debe revisar lo referente a la eficiencia la cual se define como “el grado en que el software emplea en forma optima los recursos del sistema”

2.19.5 Mantenibilidad

En la puntuación de la Mantenibilidad, se desarrolla preguntas estas están contestadas por el desarrollador del sistema y no por los usuarios.

2.19.6 Portabilidad

Para la puntuación de la Portabilidad, se desarrolla preguntas estas están contestadas por el desarrollador del sistema y no por los usuarios.



3.1 Introducción

En este capítulo se presenta el desarrollo del proyecto en el cual necesariamente se tiene que seguir y ejecutar un conjunto de actividades que se encuentran en el proceso de desarrollo que se ha optado para emplear, en el análisis y diseño de este sistema se hará bajo la metodología RUP que utiliza como herramienta de descripción al UML, con el objetivo de producir software de alta calidad que cumpla con los requerimientos de los usuarios, dentro de una planificación que cubra el ciclo de vida del desarrollo de software.

3.1.1. Análisis y situación actual

El Servicio Nacional de Meteorología en Hidrología (SENAMHI) está destinado a realizar la elaboración de venta de certificaciones de datos, meteorológicos e hidrológicos en base a las proformas la cual ellos emiten a través de solicitudes realizadas por los clientes, para así tener conocimiento de la oferta (productos), demanda (clientes) que tiene la institución.

3.2 Fase de Inicio

La fase inicial se centra en el informe preliminar de los requerimientos del Sistema además de la descripción de las áreas involucradas.

En esta fase se analizó todos los problemas que existían en las diferentes áreas que son motivos de nuestro estudio, luego se hizo un análisis de los requerimientos de los usuarios mediante entrevistas esencialmente con los usuarios y encargados de la institución.

3.2.1. Requerimientos del Sistema

Se tiene dos tipos de requerimientos funcionales y no funcionales.

3.2.1.1 Requerimientos Funcionales

Requerimientos Funcionales
Poseer una estructura de cubos de datos en el cual se debe poder configurar los elementos de datos, esta estructura realizada a través de dimensiones.
Actualizar la información de los clientes que realizan las solicitudes de certificaciones en la institución.
Generar la Proformas que solicitan los clientes a la institución
Registrar las Oficinas Regional de la institución para administrar las ventas de cada una de ellas.
Realizar el registro de los producto que la institución ofrece parámetros, productos especializados y servicios.
Realizar el registro y control de las ventas
Realizar y registrar las proformas que se solicitan.
Obtener reportes estadísticos de ventas (ofertas), solicitudes (demandas) de la institución

Tabla 3.1: Requerimientos Funcionales
Fuente: [Elaboración Propia]

3.2.1.2. Requerimientos No Funcionales

Requerimientos No Funcionales
El cliente Web del sistema debe soportar los siguientes navegadores: Microsoft Internet Explorer 6.0 o superior FireFox 1.5 o superior para Linux y para Windows
El sistema debe ser compatible con Windows 2000 profesional y Windows XP
El sistema debe permitir a un usuario manejarlo sin entrenamiento previo

Tabla 3.2: Requerimientos no Funcionales
Fuente: [Elaboración Propia]

3.2.2. Descripción del Sistema

En la Tabla 3.2 observamos funciones que debe realizar el proyecto.

Ref. N°	Función	Categoría
R1	Ingreso de Usuarios al sistema	Evidente
R2	Verificación de datos	Oculto
R3	Administrar productos	Evidente
R4	Administrar y actualización de datos del cliente	Evidente
R5	Administrar oficinas regionales	Evidente
R6	Elaboración de las proformas en base a las solicitudes.	Evidente
R7	Verificación de datos registrado	Oculto
R8	Realizar la venta de certificación.	Evidente
R9	Administrar dimensiones del cubo para elaborar los reportes	Evidente
R10	Realizara los histogramas para poder ver los resultados y así poder tomar decisiones.	Oculto
R11	Realizar informe de solicitudes (demandas)	Oculto
R12	Realizar informe de proformas (ofertas)	Oculto
R13	Realizar Informe de ventas	Oculto

Tabla 3.3: Funciones del Sistema
Fuente: [Elaboración Propia]

3.3. Fase de Elaboración

Analizar el problema y el modelo completo del negocio para establecer el dominio, a demás construir la arquitectura del proyecto.


3.3.1 Modelado del Negocio

A través del modelo de negocios hallaremos la técnica para comprender los procesos de organización del entorno. El modelo de negocios describe el proceso en términos de casos de uso y actores que intervienes en el contexto del sistema que corresponde al proceso de usuarios del mismo. El modelo de negocios esta detallado e identifica los procesos más importantes del contexto del sistema, describiendo procesos exactos relacionados con los actores.

El modelo del negocio será representado por el diagrama de casos de uso general en donde se encuentra las solicitudes de los usuarios y las respuestas que brindara el nuevo sistema.

3.3.2 Descripción de los actores

Los actores ó usuarios del sistema los cuales interactúan con el sistema está conformado por lo siguiente:

ACTORES	CLASIFICACION	DESCRIPCION
 Administrador	Nivel 1	Este actor tiene acceso a todos los Módulos del sistema el usuario puede adicionar, Eliminar y Modificar a la vez administrar los productos, clientes, oficinas regionales, productos, proformas y además las ventas de la proformas elaboradas y así poder realizar las consultas de




		los procesos estadísticos. Por otra parte podrá manejar los niveles de usuario y asignarles el mismo al personal de la institución.
 Recepcionista	Nivel 3	Este actor está encargado de registrar las solicitudes de proformas de los clientes que llegan a la institución para la elaboración de la misma
 Encargado de Proformas	Nivel 2	Este actor está encargado de realizar las proformas solicitadas por los clientes para después poder realizar la venta de la proforma.
 Encargado de ventas	Nivel 2	Este actor está encargado del registro de las ventas de certificaciones que se han realizado en la institución sobre las proformas elaboradas

Tabla 3.4: Descripción de actores
Fuente: [Elaboración Propia]

3.3.3. Identificación de Casos de Uso

Los casos de uso nos ayudan a ver los procesos necesarios para poder realizar el sistema.

DIAGRAMA DE CASO DE USO DEL SISTEMA



Figura 3.1: Diagrama de Casos del Sistema
Fuente: [Elaboración Propia]

DIAGRAMA DE CASOS DE USO INICIAR SESION

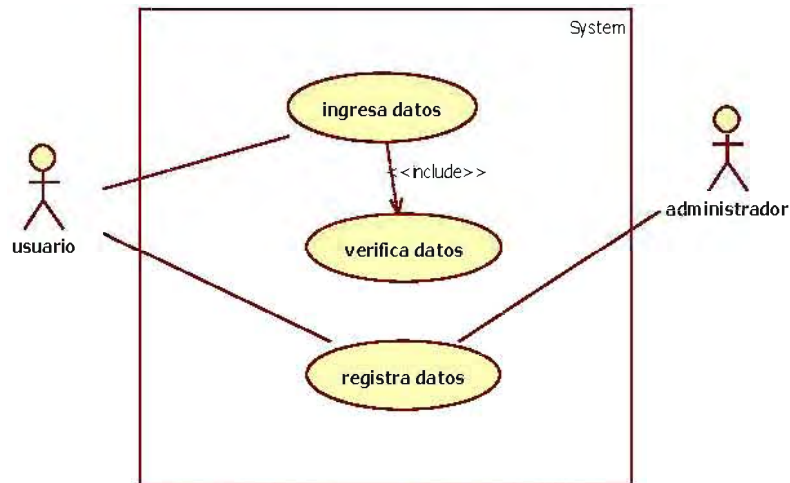


Figura 3.2: Diagrama de Casos de uso Iniciar Sesión
Fuente: [Elaboración Propia]

Nombre:	Iniciar sesión
Descripción:	Ingreso de datos para poder acceder al sistema
Actores:	Administrador, Usuario
Precondiciones	Los Datos del Usuario deben estar registrados en el sistema
Flujo de eventos	
Actores	Sistema
1. El Usuario ingresa al sistema	2. Verifica si los datos están correctos
	3. datos correctos
	4. ingresa al sistema

Tabla 3.5 Ingreso al sistema

Fuente [Elaboración Propia]

Nombre:	Registro Datos
Descripción:	Registro de datos de un nuevo usuario al sistema
Actores:	Administrador
Precondiciones	Los datos del Administrador deben estar correctos
Flujo de eventos	
Actores	Sistema
1. Administrador ingresa al sistema	2. Verifica si los datos están correcto
	3. El sistema muestra un mensaje de bienvenida
4. Administrador selecciona nuevo usuario	
	5. el sistema devuelve una ventana de nuevo Usuario
6. Administrador ingresa datos del usuario	
	7. almacena datos del nuevo usuario.
8. Administrador selecciona modificar datos.	
	9. sistema devuelve ventana para modificar los datos.
	10. sistema actualiza datos
11. Administrador selecciona eliminar usuarios del sistema	
	12. sistema elimina usuario
13. Administrador sale del sistema.	

Tabla 3.6 Registro de datos del Usuario

Fuente [Elaboración Propia]

ADMINISTRAR PRODUCTO

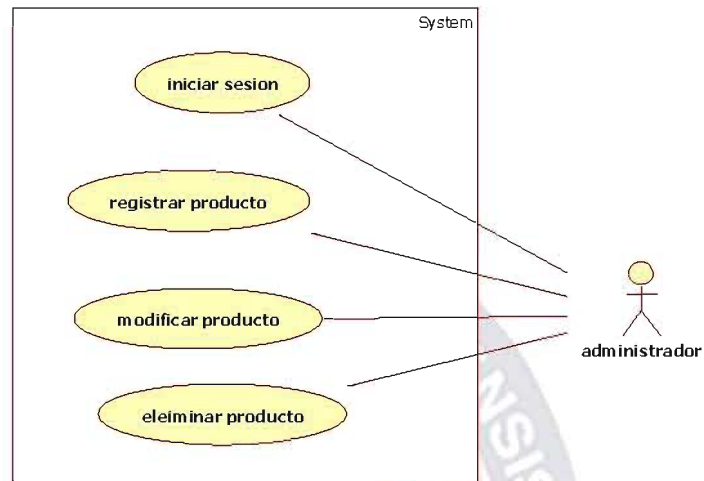


Figura 3.3: Diagrama de Casos Administrar Producto
Fuente: [Elaboración Propia]

Nombre:	Administrar Producto	
Descripción:	Registro de un producto al sistema	
Actores:	Administrador	
Precondiciones	Los datos del usuario deben estar correctos, el producto no debe existir en el sistema	
Flujo de eventos		
Actores	Sistema	
1. Administrador ingresa al sistema	2. Verificar si los datos están correctos	
	3. El sistema muestra un mensaje de bienvenida	
4. Administrador selecciona nuevo producto		
	5. el sistema devuelve una ventana de nuevo producto	
6. Administrador ingresa datos del producto		
	7. almacena datos del nuevo producto	
8. Administrador selecciona		

modificar producto	
	9. sistema devuelve ventana para editar datos del producto.
10. Administrador selecciona eliminar producto	
	11. sistema elimina producto
12. salir	

Tabla 3.7 Administrar Producto

Fuente [Elaboración Propia]

ADMINISTRAR CLIENTE

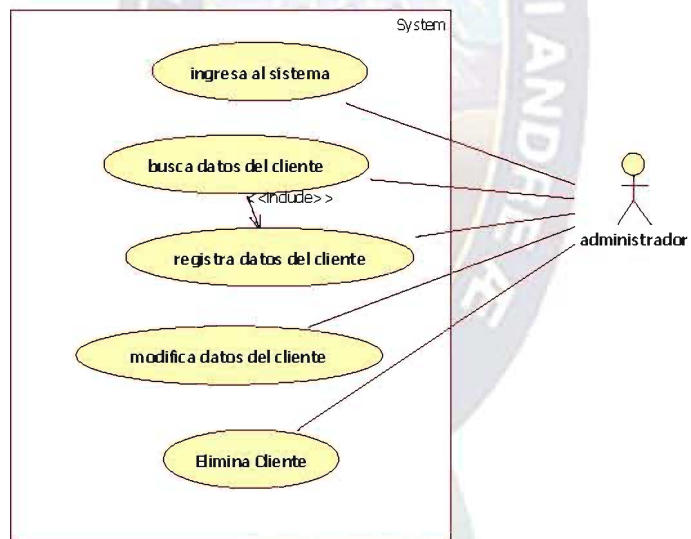


Figura 3.4: Diagrama de Casos Administrar Cliente

Fuente: [Elaboración Propia]

Nombre:	Administrar Cliente
Descripción:	Registro de un cliente al sistema
Actores:	Administrador
Precondiciones	Los datos del usuario deben estar correctos, el cliente no debe existir en el sistema
Flujo de eventos	
Actores	Sistema
1. Administrador ingresa al sistema	2. Verifica si los datos están correcto
	3. El sistema muestra un mensaje de bienvenida
4. Administrador busca al cliente introduce los datos al sistema	
	5.El sistema realiza búsqueda del Cliente 6. Encuentra cliente 7. Cliente no encontrado devuelve mensaje.
8. administrador selecciona nuevo cliente 9. Administrador ingresa datos del cliente	
	10. Almacena datos del nuevo cliente
11.Administrador selecciona modificar cliente	
	12. Sistema muestra ventana para editar cliente
13. Administrador selecciona eliminar cliente	
	14. sistema elimina cliente.
15. Salir	

Tabla 3.8. Administrar cliente

Fuente [Elaboración Propia]

PROFORMA



Figura 3.5: Diagrama de Casos Generar Proforma
Fuente: [Elaboración Propia]

Nombre:	Elaborar Proforma
Descripción:	Verificar datos del cliente para realizar la proforma que se solicito por el cliente
Actores:	Administrador, usuario
Precondiciones	Los datos de los usuarios deben estar correctos.
Flujo de eventos	
Actores	Sistema
1. Administrador ingresa al sistema	2. Verifica si los datos están correctos
	3. El sistema muestra un mensaje de bienvenida
4. Administrador selecciona cliente	
	5. el sistema devuelve lista de clientes
6. Administrador ingresa datos de búsqueda del cliente para realizar la proforma que se solicito	
	7. sistema devuelve datos encontrados
8. Administrador selecciona realizar proforma para el cliente seleccionado	
	9. sistema obtienes datos del cliente
10. Administrador ingresa datos de inicio y de fin en el calendario	
11. administrador selecciona estación, selecciona producto, selecciona detalle de lo precios, por ultimo ingresa cantidad y por ultimo agrega el producto	
	12. sistema al seleccionar estación automáticamente habilita el producto una vez seleccionado automáticamente se habilita os precios del producto seleccionado, al seleccionar le producto el sistema habilitas los tipos de precio que existe de cada uno de los detalles al seleccionar el detalle se habilita el precio, y por ultimo ingresa cantidad de datos
13. Administrador selecciona agregar.	

	15. sistema agrega detalle de la proforma y pregunta si quiere nuevo producto o genera proforma
17. Administrador puede agregar otra proforma o generar la proforma. 18 Salir.	16. sistema muestra agregar detalle y genera proforma.

Tabla 3.9 Elaborar Proforma

Fuente [Elaboración Propia]



3.3.4 Diagrama de clases

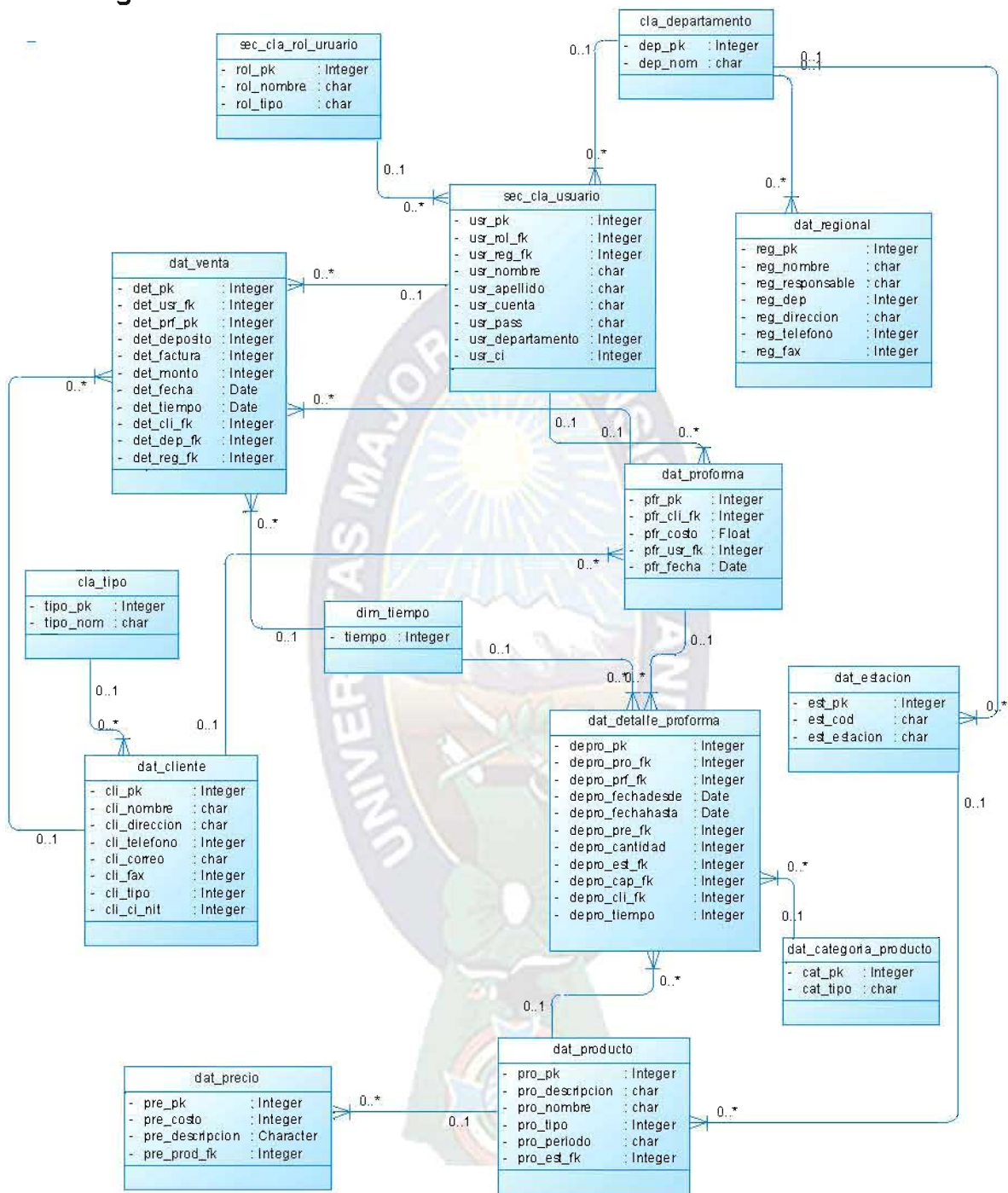


Figura: 3.6 Diagrama de Clases

Fuente: [Elaboración Propia]

3.3.4.1 Descripción de Tablas del Diagrama de Clases

dat_detalle_proforma	
- depro_pk	: Integer
- depro_pro_fk	: Integer
- depro_prf_fk	: Integer
- depro_fechadesde	: Date
- depro_fechahasta	: Date
- depro_pre_fk	: Integer
- depro_cantidad	: Integer
- depro_est_fk	: Integer
- depro_cap_fk	: Integer
- depro_cli_fk	: Integer
- depro_tiempo	: Integer

dat_venta	
- det_pk	: Integer
- det_usr_fk	: Integer
- det_prf_fk	: Integer
- det_deposito	: Integer
- det_factura	: Integer
- det_monto	: Integer
- det_fecha	: Date
- det_tiempo	: Date
- det_cli_fk	: Integer
- det_dep_fk	: Integer
- det_reg_fk	: Integer

sec_cla_usuario	
- usr_pk	: Integer
- usr_rol_fk	: Integer
- usr_reg_fk	: Integer
- usr_nombre	: char
- usr_apellido	: char
- usr_cuenta	: char
- usr_pass	: char
- usr_departamento	: Integer
- usr_ci	: Integer

dat_cliente	
- cli_pk	: Integer
- cli_nombre	: char
- cli_direccion	: char
- cli_telefono	: Integer
- cli_correo	: char
- cli_fax	: Integer
- cli_tipo	: Integer
- cli_ci_nit	: Integer

dat_regional	
- reg_pk	: Integer
- reg_nombre	: char
- reg_responsable	: char
- reg_dep	: Integer
- reg_direccion	: char
- reg_telefono	: Integer
- reg_fax	: Integer

dat_producto	
- pro_pk	: Integer
- pro_descripcion	: char
- pro_nombre	: char
- pro_tipo	: Integer
- pro_periodo	: char
- pro_est_fk	: Integer

dat_profoma	
- pfr_pk	: Integer
- pfr_cli_fk	: Integer
- pfr_costo	: Float
- pfr_usr_fk	: Integer
- pfr_fecha	: Date

dat_precio	
- pre_pk	: Integer
- pre_costo	: Integer
- pre_descripcion	: Character
- pre_prod_fk	: Integer

dat_estacion	
- est_pk	: Integer
- est_cod	: char
- est_estacion	: char

sec_cla_rol_usuario	
- rol_pk	: Integer
- rol_nombre	: char
- rol_tipo	: char

dat_categoria_producto	
- cat_pk	: Integer
- cat_tipo	: char

cla_departamento	
- dep_pk	: Integer
- dep_nom	: char

dim_tiempo	
- tiempo	: Integer

cla_tipo	
- tipo_pk	: Integer
- tipo_nom	: char

3.3.5 Diagrama de Secuencia

En el diagrama de secuencia describiremos el comportamiento dinámico del sistema de información haciendo énfasis a la secuencia de los mensajes intercambiados

INICIAR SESION

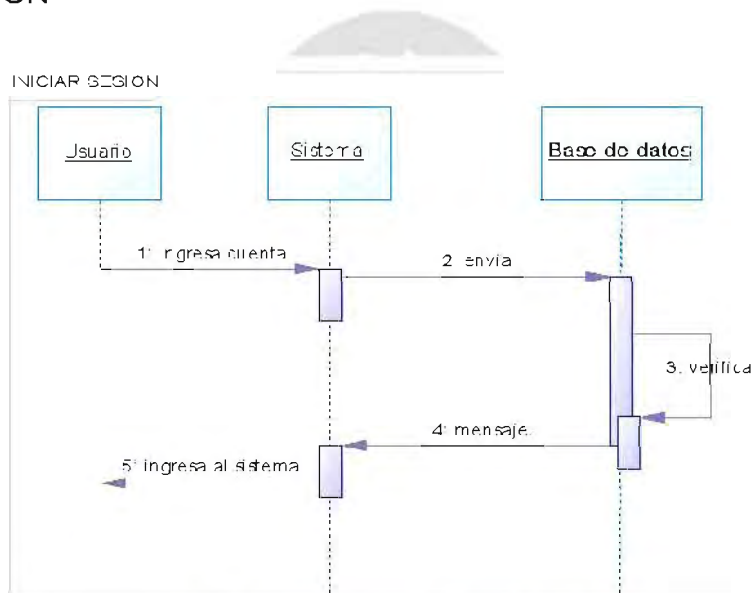


Figura 3.7: Diagrama de Secuencia Iniciar Sesión
Fuente: [Elaboración Propia]

ADMINISTRAR CLIENTE

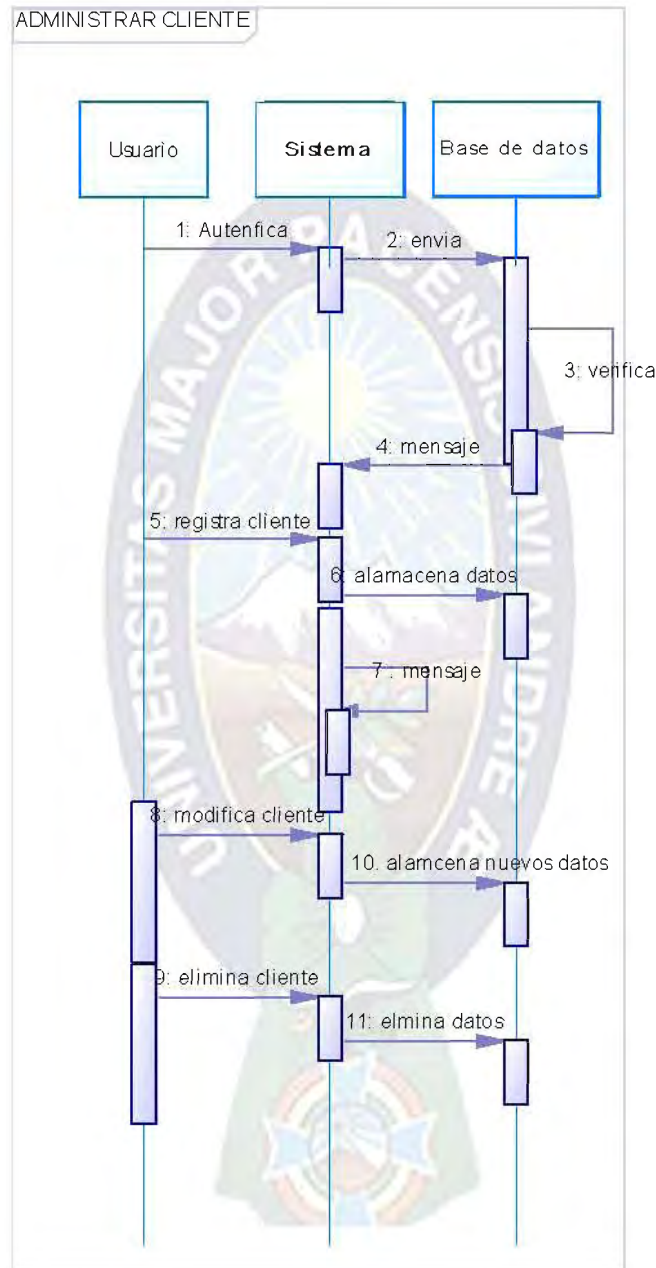


Figura 3.8: Diagrama de Secuencia Administrar Cliente
Fuente: [Elaboración Propia]

ADMINISTRAR PRODUCTO

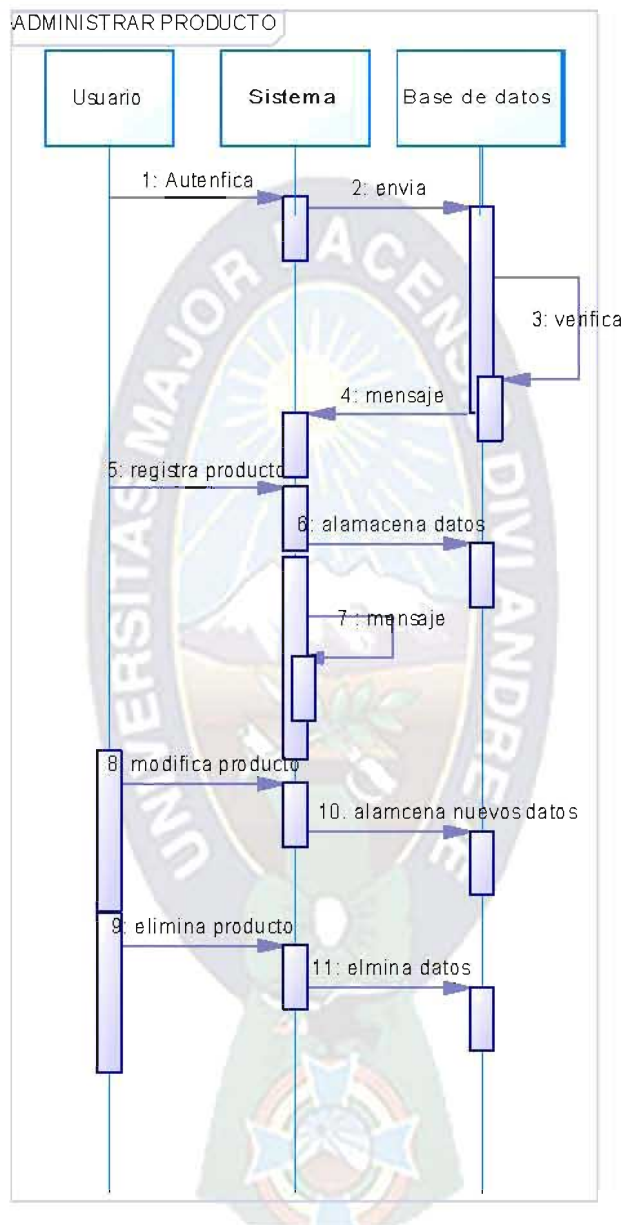


Figura 3.9: Diagrama de Secuencia Administrar Producto
Fuente: [Elaboración Propia]

PROFORMA

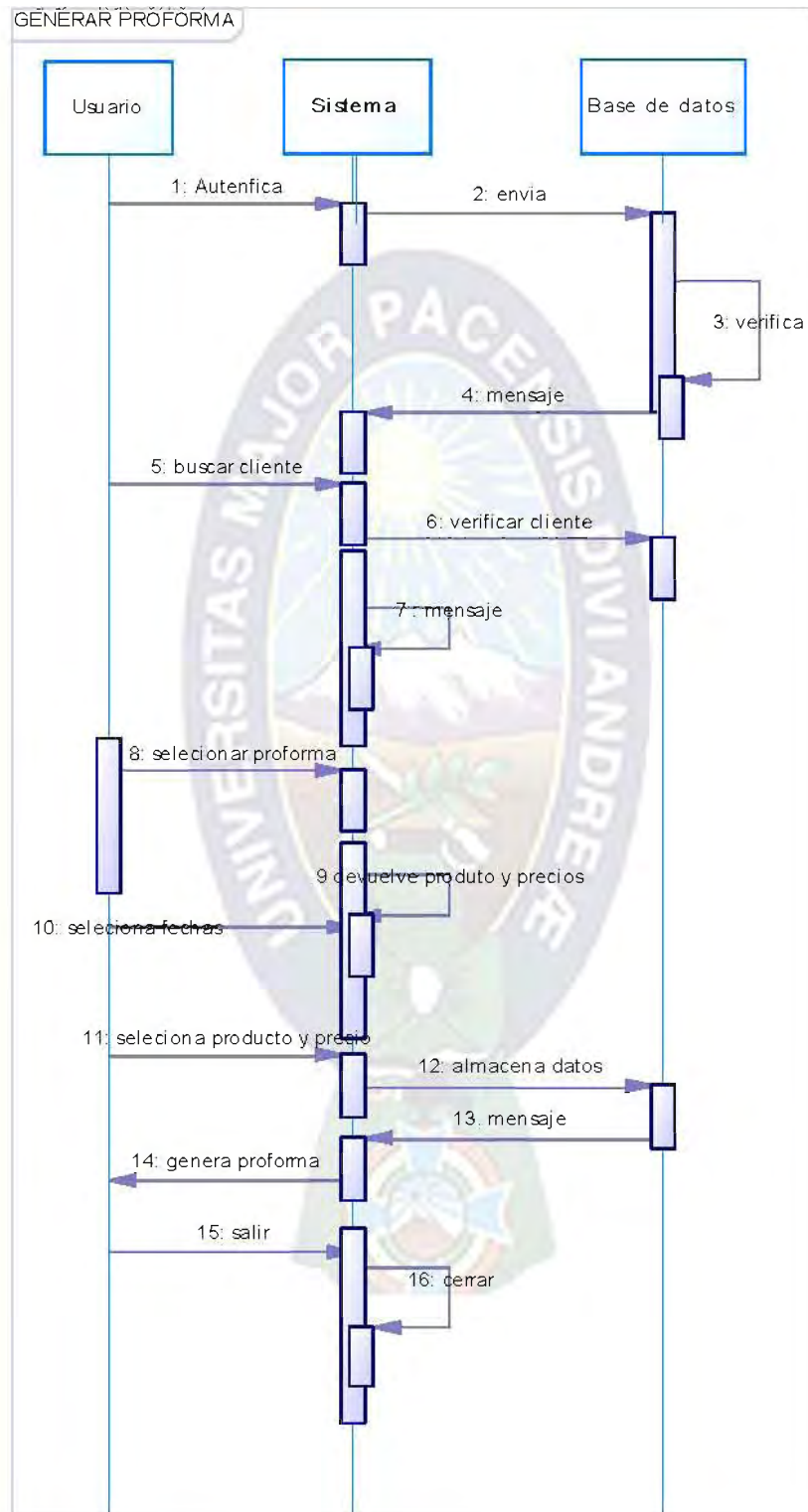


Figura 3.10: Diagrama de Secuencia Elaborar Proforma
Fuente: [Elaboración Propia]

ADMINISTRAR DIMENSION

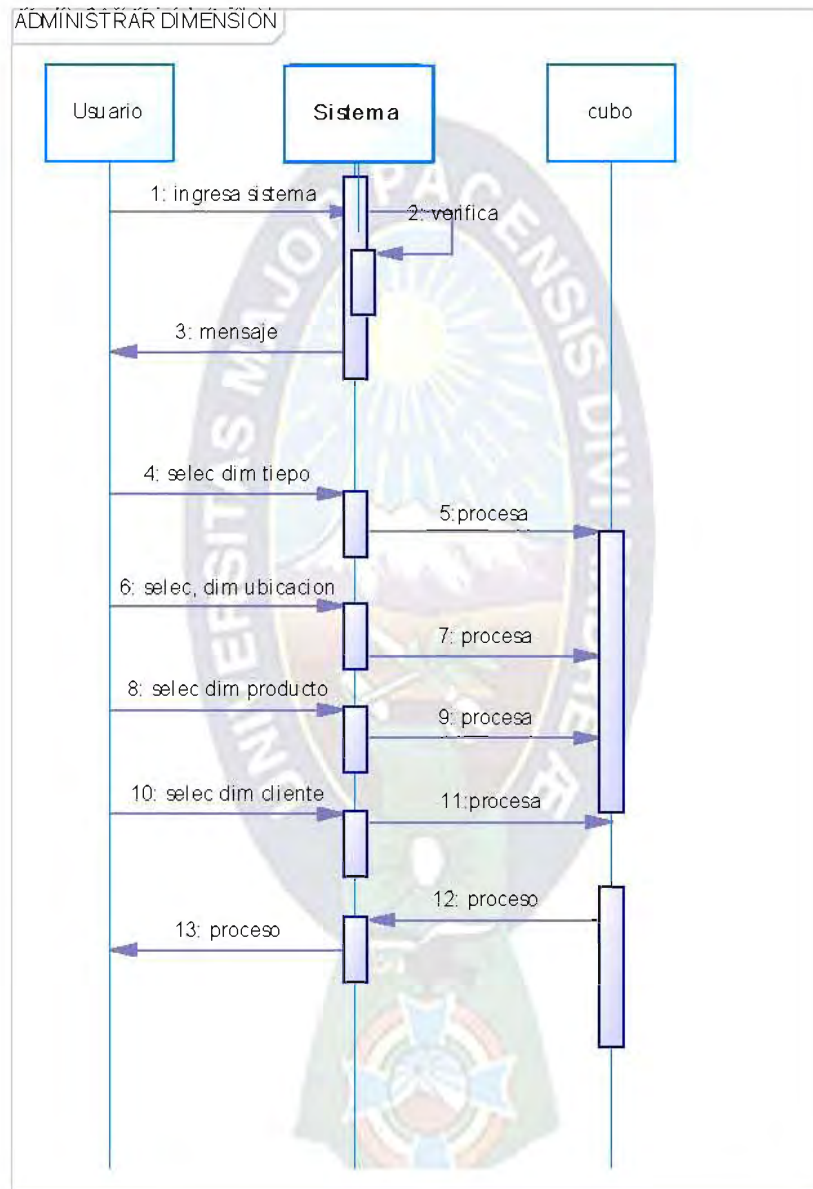


Figura 3.11: Diagrama de Secuencia Administrar Dimensión
Fuente: [Elaboración Propia]

3.3.6 Diagrama de Estados

Un diagrama de estado presenta el ciclo de vida de un objeto los eventos que le ocurren, sus transmisiones de cada uno de los eventos que le corresponde a cada uno de ellos.

INICIAR SESION

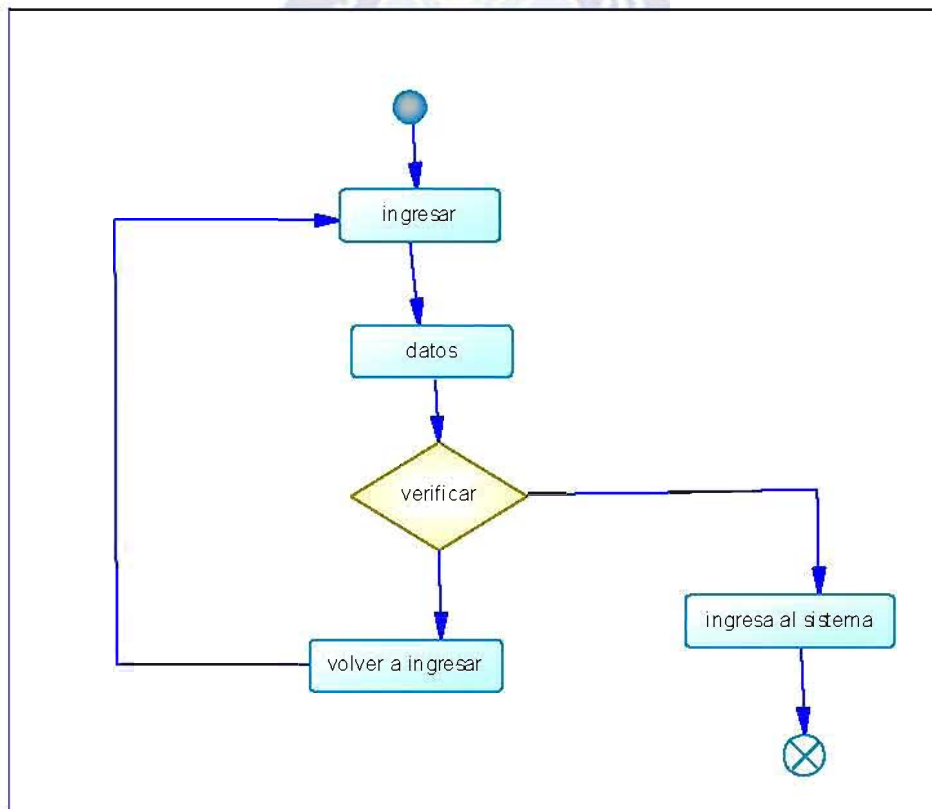


Figura 3.12: Diagrama de Estados Iniciar Sesión
Fuente: [Elaboración Propia]

ADMINISTRAR CLIENTE

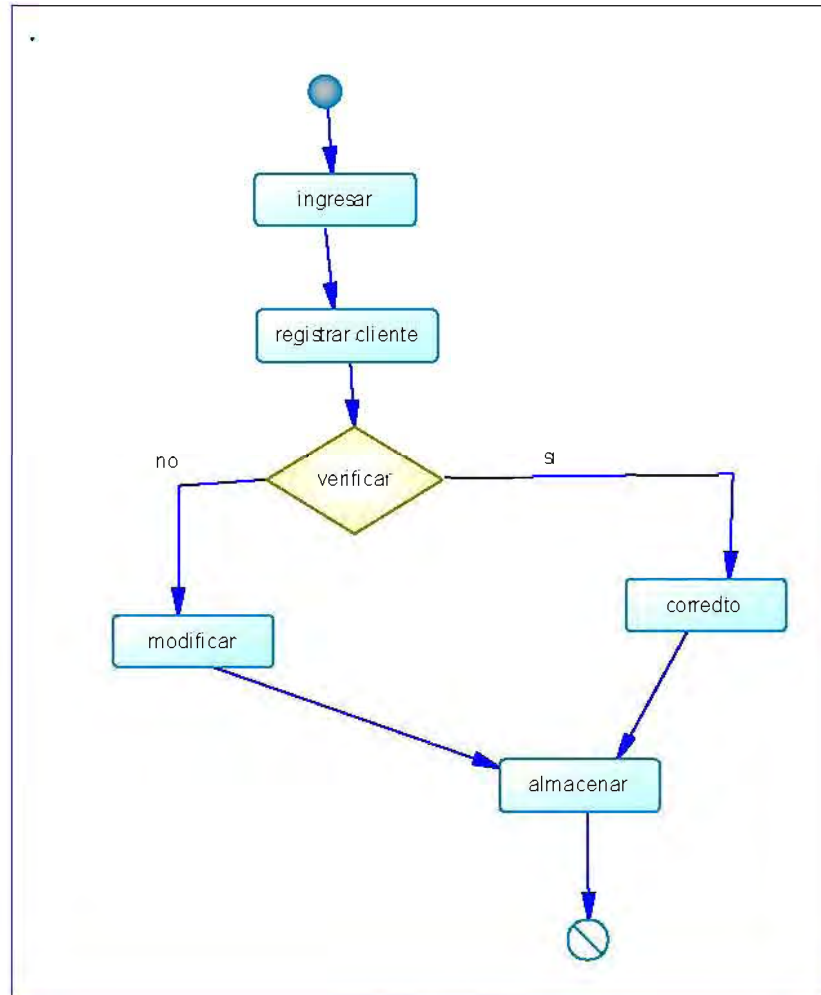


Figura 3.13: Diagrama de Estados Administrar Cliente
Fuente: [Elaboración Propia]

ADMINISTRAR PRODUCTO

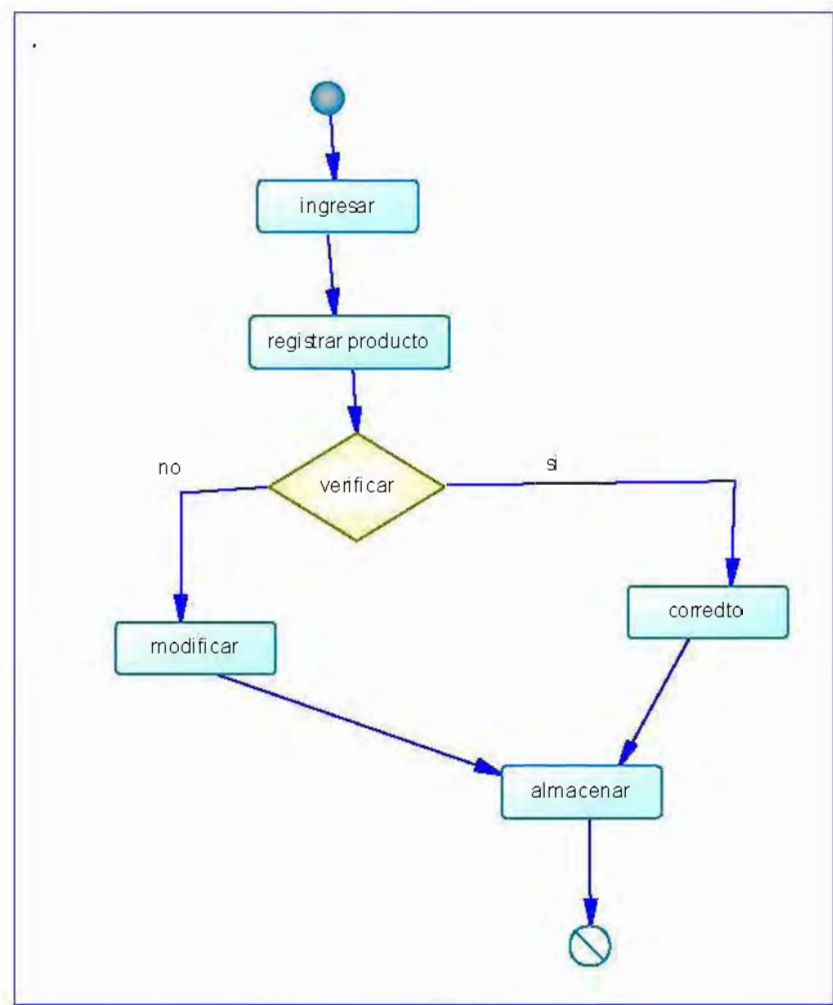


Figura 3.14: Diagrama de Estados Administrar producto
Fuente: [Elaboración Propia]

PROFORMA

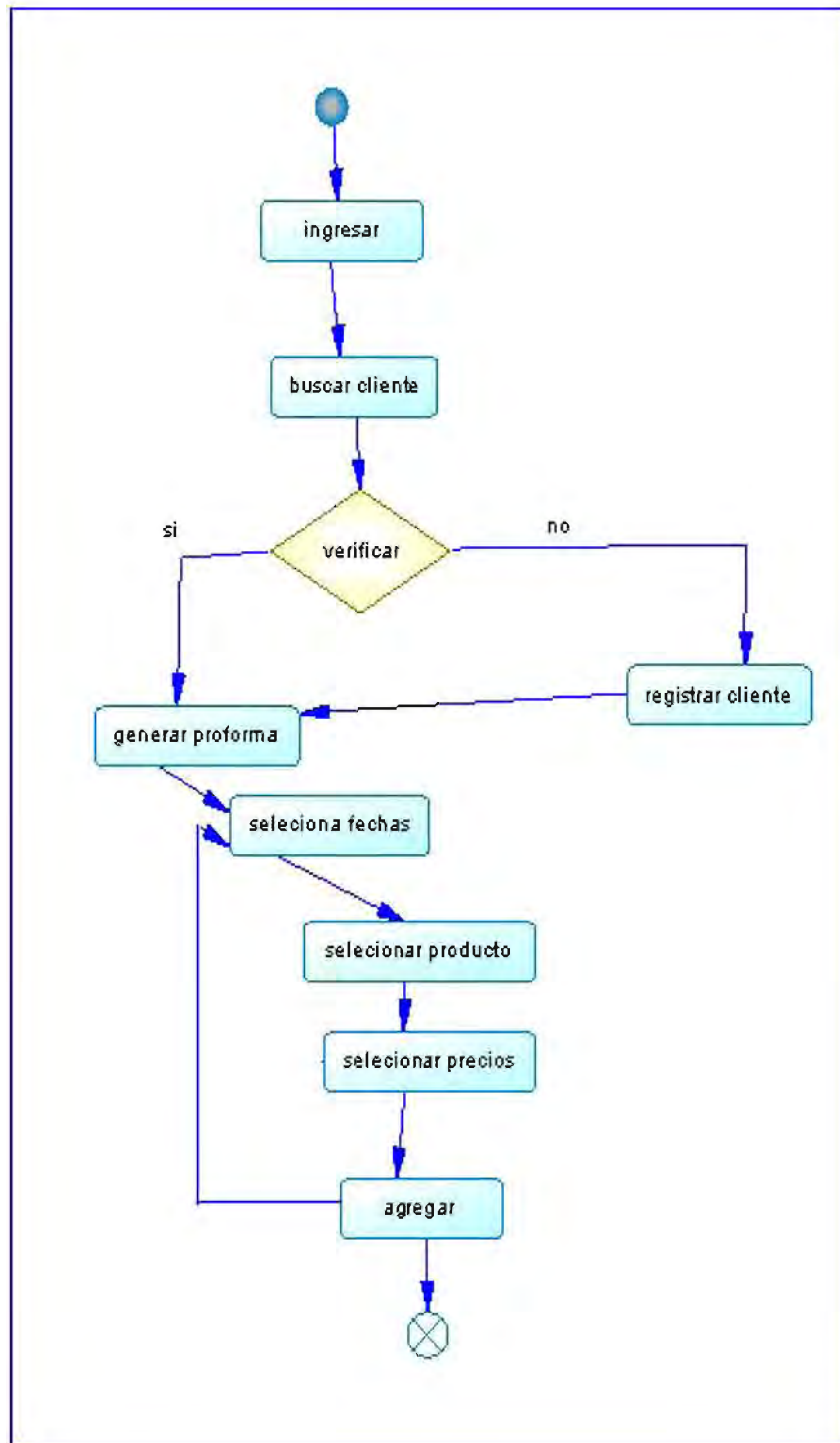


Figura 3.15: Diagrama de Estados Elaborar proforma
Fuente: [Elaboración Propia]

3.3.7 Diagrama de Colaboración

INICIAR SESION

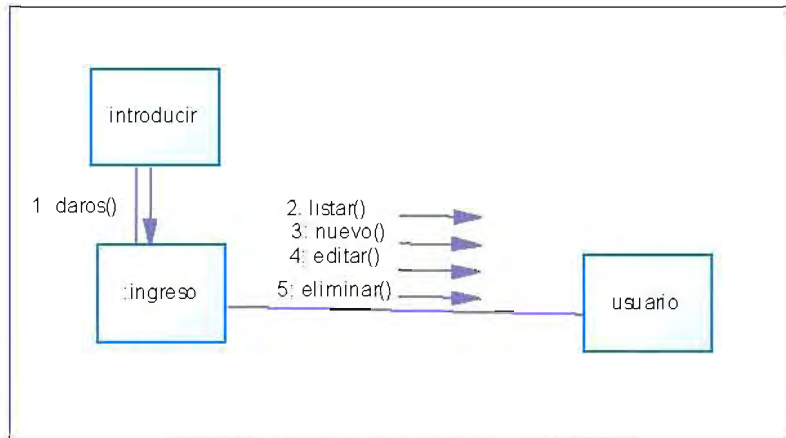


Figura 3.16: Diagrama Colaboración Iniciar Sesión
Fuente: [Elaboración Propia]

ADMINISTRAR CLIENTE

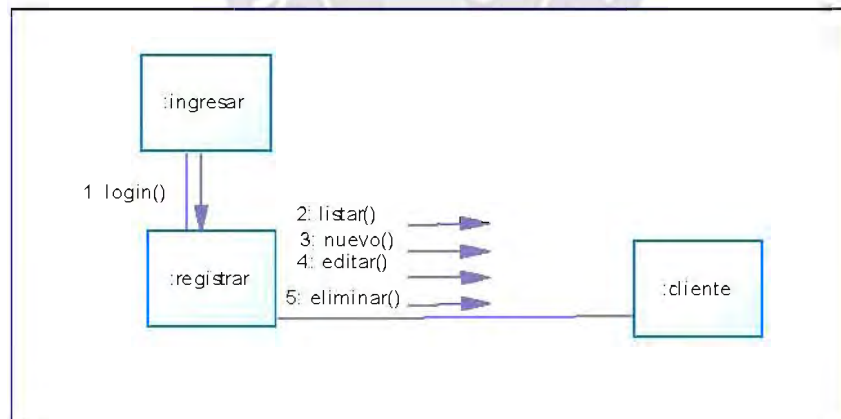


Figura 3.17: Diagrama Colaboración Administrar Cliente
Fuente: [Elaboración Propia]

ADMINISTRAR PRODUCTO

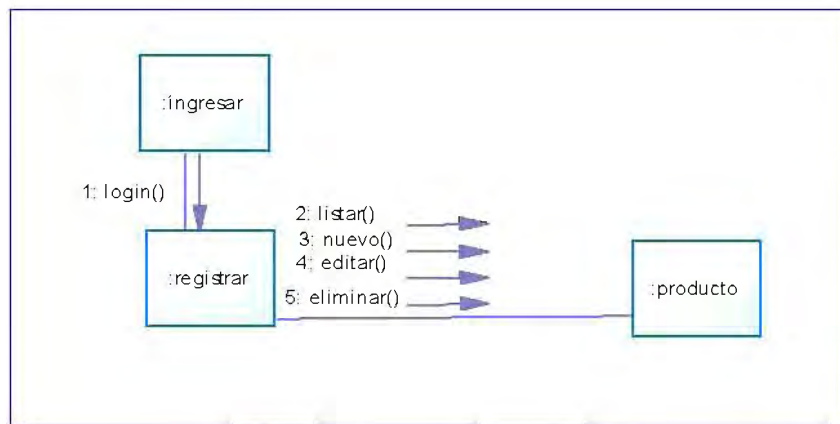


Figura 3.18: Diagrama Colaboración Administrar Producto

Fuente: [Elaboración Propia]

PROFORMA

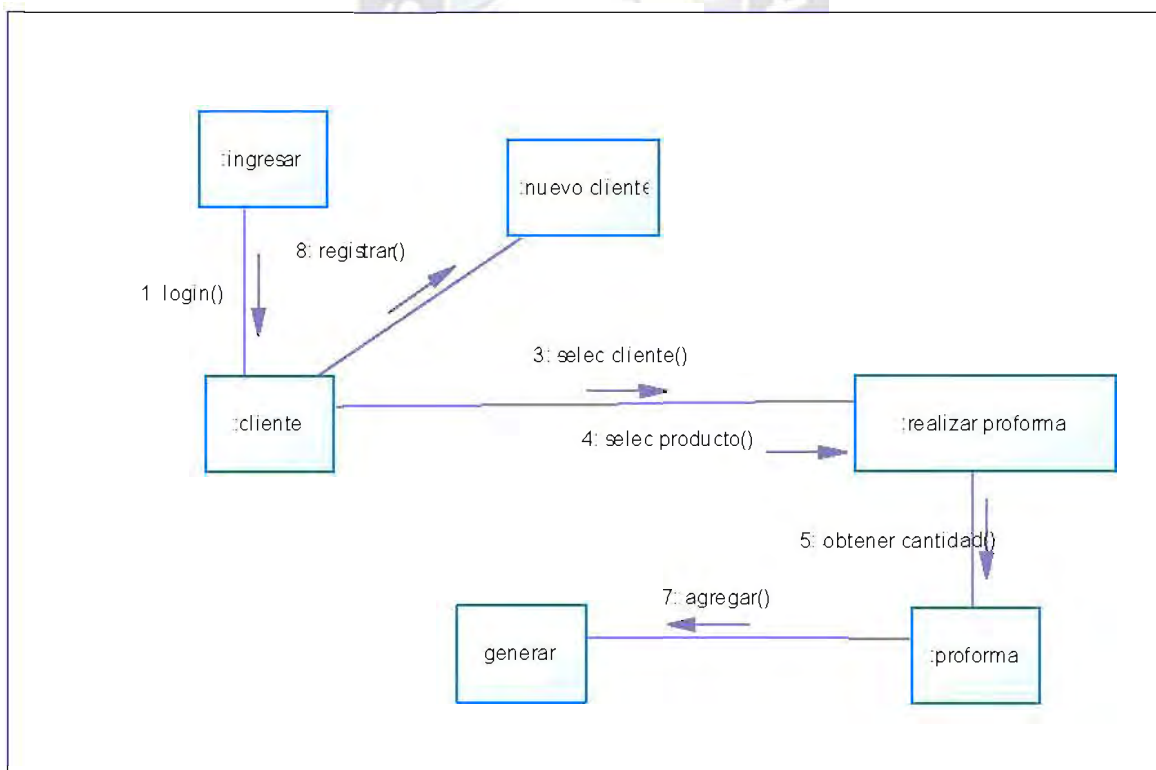


Figura 3.19: Diagrama Colaboración proceso proforma

Fuente: [Elaboración Propia]

3.3.8 Cubo

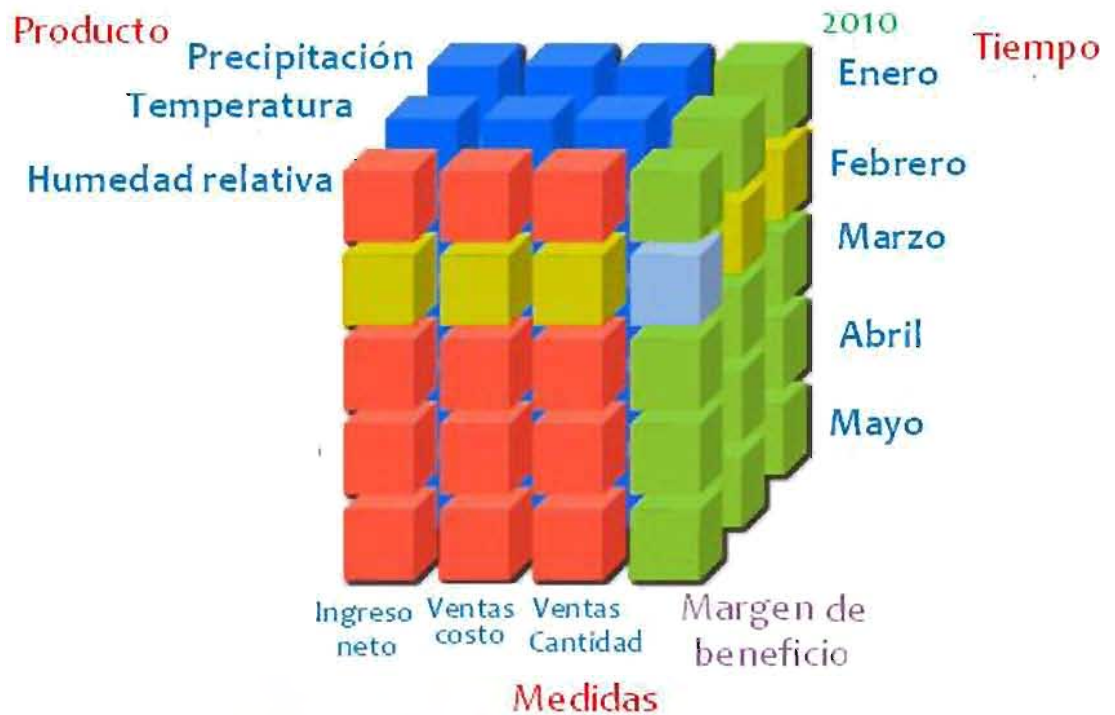


Figura 3.20: Cubo (producto, medidas, tiempo)
Fuente: [Elaboración Propia]

La figura 3.20: representa algunas dimensiones del Cubo como podemos ver se está tomando en cuenta la dimensión del producto, las medidas de las ventas que se realizo de los productos y el tiempo en el que se las realizo las ventas de los productos.

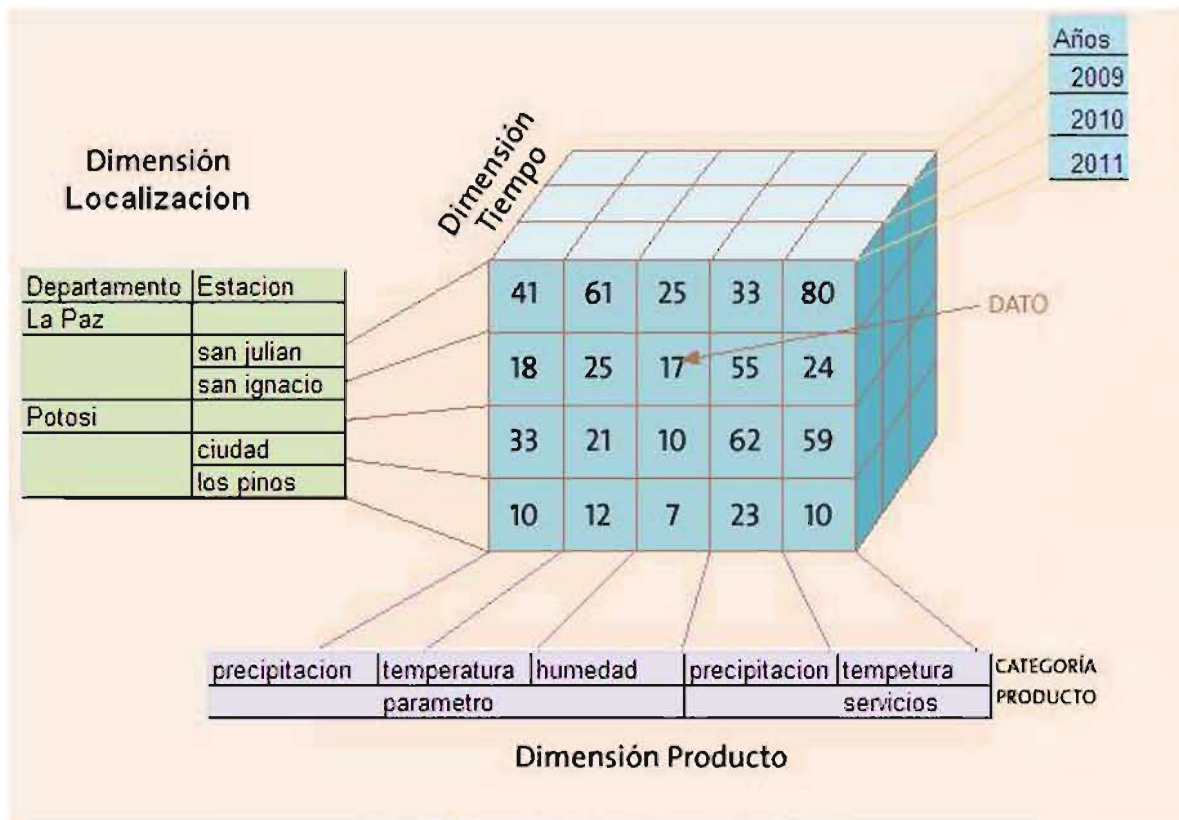


Figura 3.21: Cubo (categoría producto, localización, tiempo)
Fuente: [Elaboración Propia]

La figura 3.21: representa algunas dimensiones del Cubo como podemos ver se está tomando en cuenta la dimensión de la categoría de los productos, la localización de donde se realizaron las ventas, la cantidad de los datos y el tiempo en la que se realizo

3.4 Fase de Construcción

Durante la fase de construcción se tiene más énfasis en el diseño y la implementación que es el flujo fundamental en donde se modela el sistema y se encuentra su forma para que soporte todos los requerimientos, y desarrollo del proyecto tomando como una abstracción para la implementación del sistema.

3.4.1 Diagrama De Componentes

El diagrama de componentes muestra de manifiesto los componentes, con sus interfaces bien definidas y la interacción entre estos y los casos de uso.

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones.

DIAGRAMA DE COMPONENTES

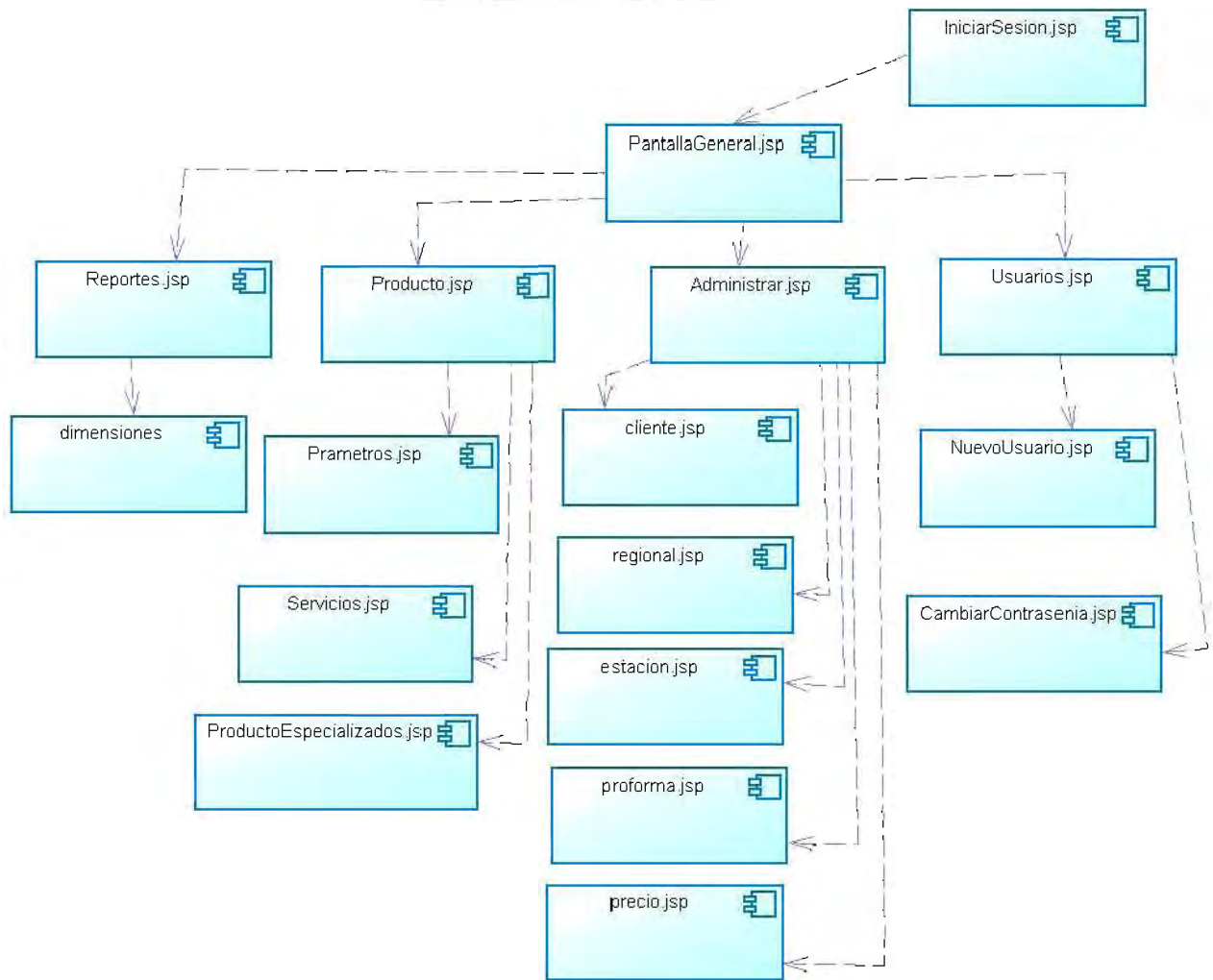


Figura 3.22: Diagrama de Componentes
Fuente: [Elaboración Propia]

3.4.2. Diagrama De Despliegue

Aquí es donde se muestra los nodos del sistema, dentro de los cuales se encuentran, los componentes que se encuentran en los nodos.

Los diagramas de despliegue muestran la disposición física de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos.

DIAGRAMA DE DESPLIEGUE

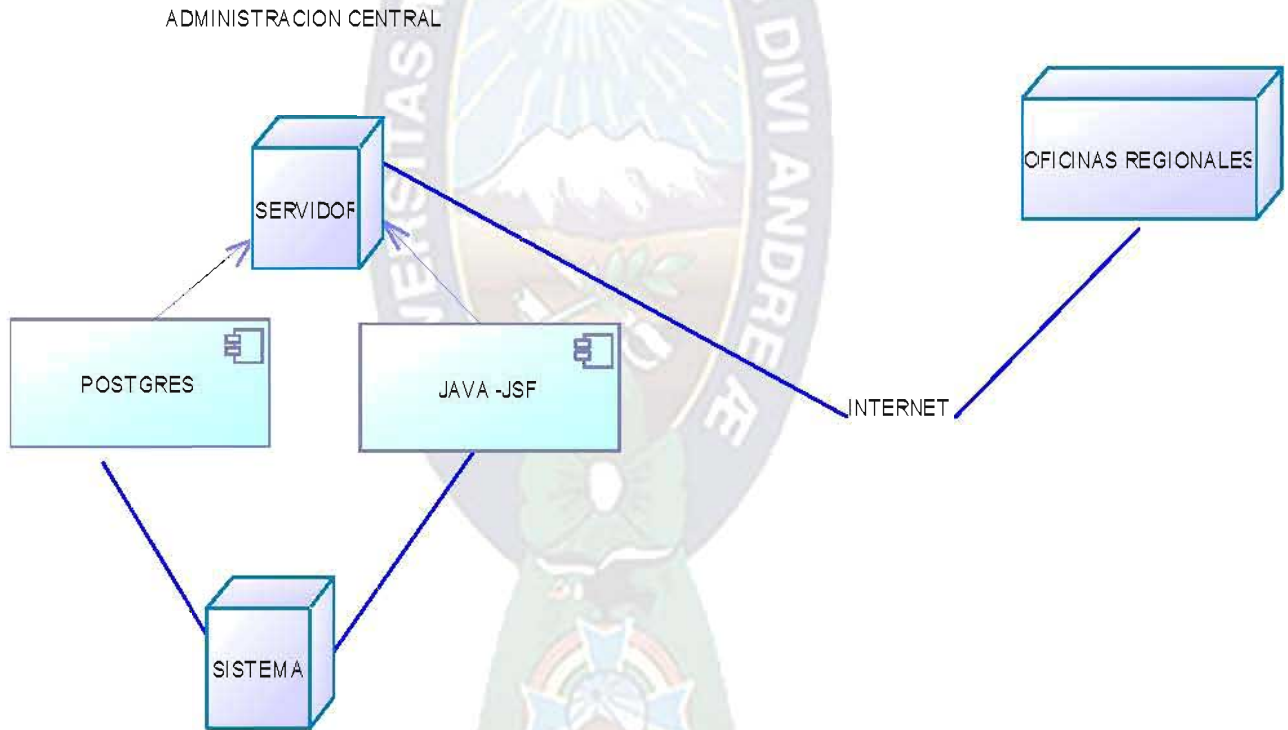


Figura 3.23: Diagrama Despliegue
Fuente: [Elaboración Propia]

3.4.3. Arquitectura del Sistema

ARQUITECTURA DEL SISTEMA

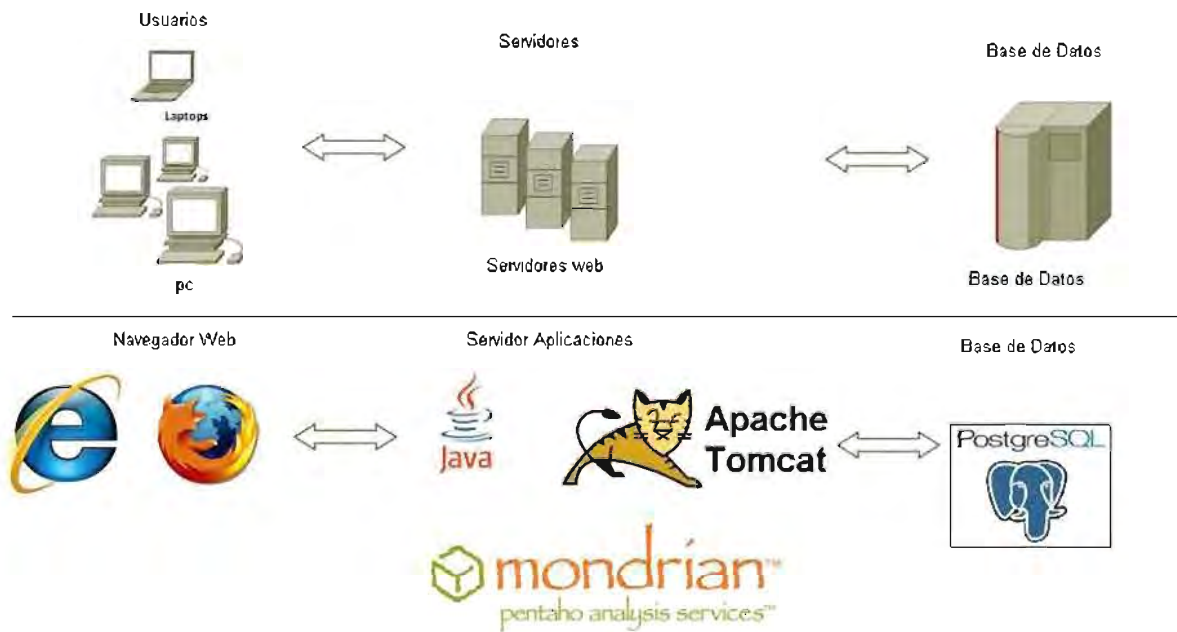


Figura 3.24: Arquitectura del Sistema
Fuente: [Elaboración Propia]

En la fase de construcción mostramos las pantallas del sistema

3.4.4. Implementación

3.4.4.1. Ventanas del sistema

Las ventanas de usuarios están desarrolladas de un forma que presenta una interactividad y facilidad de trabajo, de tal manera que puede ser fácil de comprender.

a) Ingreso al sistema el cliente debe validar su datos de usuario contraseña en caso de que ingrese datos erróneos el sistema de mostrara el mensaje que se encuentra en la figura con letras rojas.

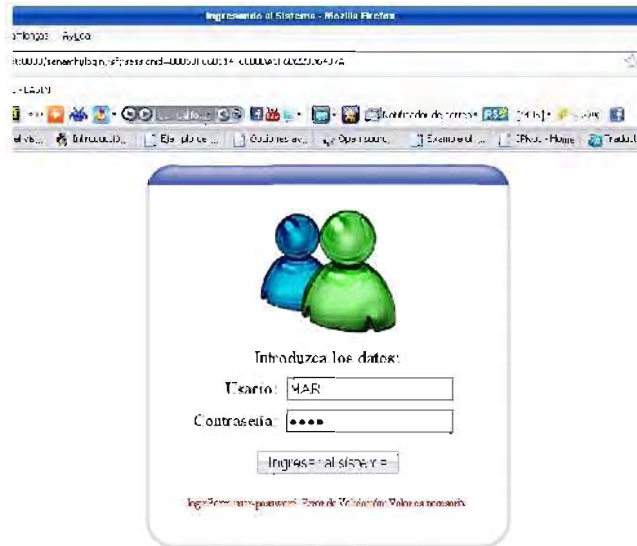


Figura 3.25: Iniciar Sesión
Fuente: [Elaboración Propia]

b) Listado de los clientes para elaborar al proforma solicitada



Figura 3.26: Lista de Clientes
Fuente: [Elaboración Propia]

C) Elaboración de la proforma



Figura 3.27: elaboración de la proforma

Fuente: [Elaboración Propia]

D) seleccionando producto para agregar a la proforma

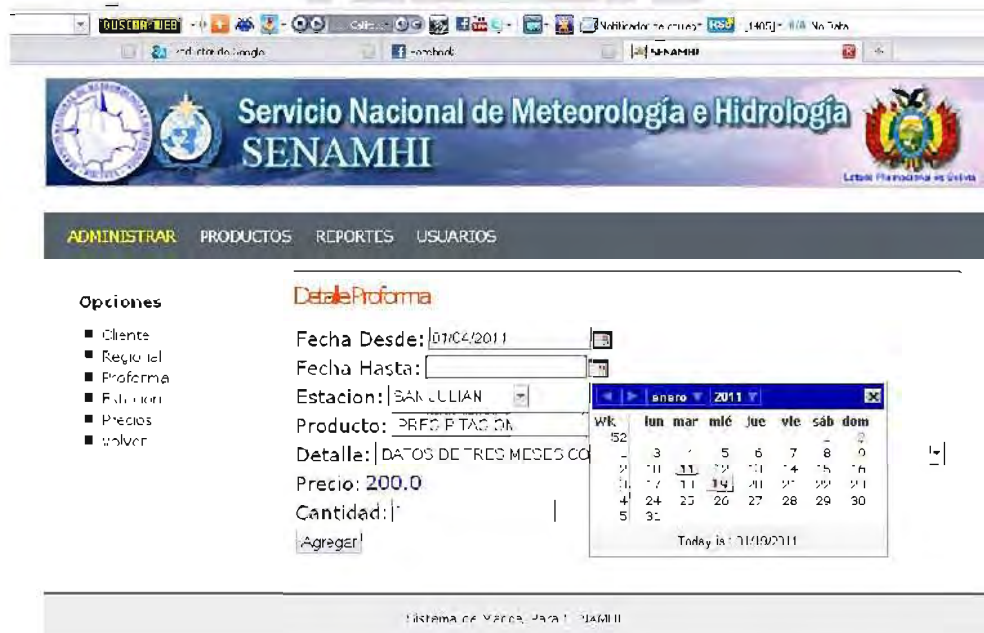


Figura 3.28: Seleccionar Producto

Fuente: [Elaboración Propia]

Profoma

Razon Social: MARBE...

Nit / CI: 8053814

Direccion: A.D.F.L. NID

• Seleccionar Producto

Fecha desde	Fecha Hasta	Producto	Estacion	Detalle	Costo	Cantidad	Sub Total
Tue Jan 04 00:00:00 BOT 2011	Tue Jan 11 00:00:00 BOT 2011	PRECIPITACION	SAN TULCAN	DATOS DE TRFS MFSFS COMPLETOS: 0 FRACCION DE MES	2.000	1	200.00
Mon Jan 03 00:00:00 BOT 2011	Mon Jan 10 00:00:00 BOT 2011	NIVELES	SAN TULCAN	NIVELES MAJUS: 00.525	00.525	2	57.2

Generar

Sistema de Mariscal para SENAMHI

Figura 3.29: Seleccionar Producto
Fuente: [Elaboración Propia]

E) Listado de proformas elaboradas

Lista de Proforma

Fecha	Razon Social	Nit / CI	Total	Usuario
2011-01-16	IVAK	256789	070.00	MARICEL
2011-01-20	IVAK	252525	127.00	MARICEL
2011-01-20	IVAK	252525	127.00	MARICEL
2011-01-20	MARBE...	6263614	872.00	MARICEL
2011-01-20	MARBE...	6263614	2.00	MARICEL
2011-01-19	MARBE...	6263614	127.00	MARICEL
2011-01-17	MARBE...	6263614	33.00	MARICEL
2011-01-10	MARBE...	6263614	2.00	MARICEL

Generar

Sistema de Mariscal para SENAMHI

Figura 3.30: Listado de Proforma
Fuente: [Elaboración Propia]

G) registro de ventas



Figura 3.31: Registro de Venta
Fuente: [Elaboración Propia]

H) Listado de Ventas



Figura 3.32: Listado de Ventas
Fuente: [Elaboración Propia]

E) Registro de producto

Sistema de Maricel para SENAMHI

Figura 3.33: Nuevo Producto

Fuente: [Elaboración Propia]

Lista de Productos

Nombre producto	Descripcion	Estacion	Tipo	codigo de producto		
NIVELES	HIDROLOGICO	ABAROA - MAURI	PARAMETRO	70	✖	📄
CAUDALES	HIDROLOGICO	ABAROA - MAURI	PARAMETRO	71	✖	📄
AFOROS SOLIDOS	HIDROLOGICO	ABAROA - MAURI	PARAMETRO	72	✖	📄
PRECIPITACION	METEOROLOGICO	ABAROA - MAURI	PARAMETRO	50	✖	📄
PRECIPITACION MAXIMA	METEOROLOGICO	ABAROA - MAURI	PARAMETRO	51	✖	📄
PRECIPITACION MINIMA	METEOROLOGICO	ABAROA - MAURI	PARAMETRO	52	✖	📄

Figura 3.34: Lista de Productos

Fuente: [Elaboración Propia]

F) Registro de Precios

Historia Marcadores Herramientas Ayuda

http://localhost:8084/senamhi/ListaPrecios.do

Curso J2EE - LAGUN Curso J2EE - LAGUN

Buscador Web

Servicio Nacional de Meteorología e Hidrología
SENAMHI
Estado Plurinacional de Bolivia

ADMINISTRAR PRODUCTOS REPORTES USUARIOS

Opciones

- Cliente
- Regional
- Provincia
- Estacion
- Precios
- Jetale

Precio de un Producto

Precio del Producto:

Precio:

Desempeño:

Sistema de Meteo del Para SENAMHI

Figura 3.35: Nuevo Precio
Fuente: [Elaboración Propia]

H) Registro de Clientes

http://localhost:8084/senamhi/ListaClientes.do

Curso J2EE - LAGUN Curso J2EE - LAGUN

Buscador Web

Servicio Nacional de Meteorología e Hidrología
SENAMHI
Estado Plurinacional de Bolivia

ADMINISTRAR PRODUCTOS REPORTES USUARIOS

Opciones

- Cliente
- Regional
- Provincia
- Estacion
- Precios
- Jetale

Clientes

Tipo Cliente:

Razon Econ:

Nit:

Direccion:

Telefono: Fax:

Correo:

Sistema de Meteo del Para SENAMHI

Figura 3.36: Registro de Clientes
Fuente: [Elaboración Propia]

l) Registro de Oficinas Regionales



The screenshot shows a web browser window displaying the SENAMHI (Servicio Nacional de Meteorología e Hidrología) website. The page title is "Oficinas Regionales". On the left, there is a navigation menu with the following options: ADMINISTRAR, PRODUCTOS, REPORTES, and USUARIOS. Below the menu, there is a section titled "Opciones" with a list of checkboxes: Cliente, Oficina Regional, Profoma, Estacion, Precios, and volver. The main content area contains a form for registering a regional office. The form fields are: Tipo Cliente (BENI), Nombre (TRINIDAD), Responsable (LUIS PHILLIPS RIVERO), Direccion (07 ZONA 12 DE ABRIL), Telefono (4820490), and Fax (4820490). There are "Aceptar" and "Cancelar" buttons at the bottom of the form. A footer bar at the bottom of the page reads "Sistema de Maricel Para SENAMHI".

Programació... Color del vis... Introducció... Ejemplo de ... Opciones av... Open sourc... Exampe of ... JPivot - Home Traductor d... JPivot Demo

Servicio Nacional de Meteorología e Hidrología
SENAMHI
Estado Plurinacional de Bolivia

ADMINISTRAR PRODUCTOS REPORTES USUARIOS

Opciones

- Cliente
- Oficina Regional
- Profoma
- Estacion
- Precios
- volver

Oficinas Regionales

Tipo Cliente: BENI
Nombre: TRINIDAD
Responsable: LUIS PHILLIPS RIVERO
Direccion: 07 ZONA 12 DE ABRIL
Telefono: 4820490 Fax: 4820490

Aceptar Cancelar

Sistema de Maricel Para SENAMHI

Figura 3.37: Registro de Oficinas Regionales
Fuente: [Elaboración Propia]

J) Reportes

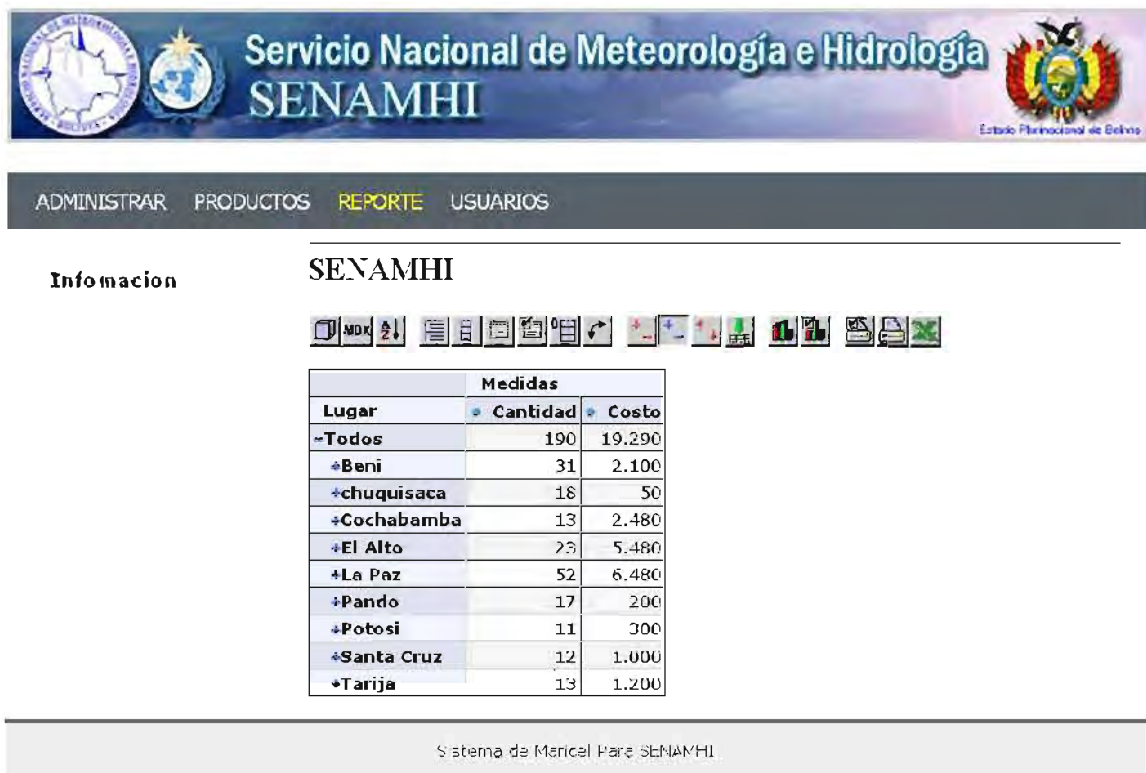


Figura 3.38: Vista inicial
Fuente: [Elaboración Propia]

Como podemos ver en la figura 3.38 el sistema está mostrando la cantidad de ventas que se han realizado en la institución en base a cada uno de los Lugares (Oficinas Regionales), donde se realizan las ventas y a la vez se está mostrando los costos o ingresos de cada uno de ellos y el total de todas las ventas.

Información

SENAMHI

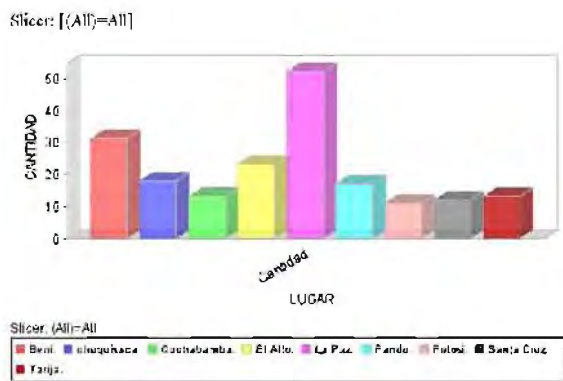


Lugar	Medidas	
	Cantidad	Costo
- Todos	190	19.290
- Beni	31	2.100
laguna suares	31	2.100
- chuquisaca	18	50
sucre	18	50
- Cochabamba	13	2.480
Av. Santa Cruz	13	2.480
- El Alto	23	5.480
12 de octubre	4	900
Av. Bolivia	19	4.580
- La Paz	57	6.480
20 de octubre	10	2.500
achocalla	19	10
Reyes Ortiz	8	900
Rosendo Gutierrez	15	2.960
- Pando	17	200

Figura 3.39: Vista desglosado el lugar
Fuente: [Elaboración Propia]

Como podemos ver en la figura 3.39 el sistema está mostrando la cantidad de las ventas que se ha realizado en la institución en base a la ubicación (Oficinas Regionales), de donde se realizaron las ventas junto a la cantidad de cada una de ellas más el costo.

Información



Sistema de Mapeo Para SENAMHI

Figura 3.40: Vista con Grafico
Fuente: [Elaboración Propia]

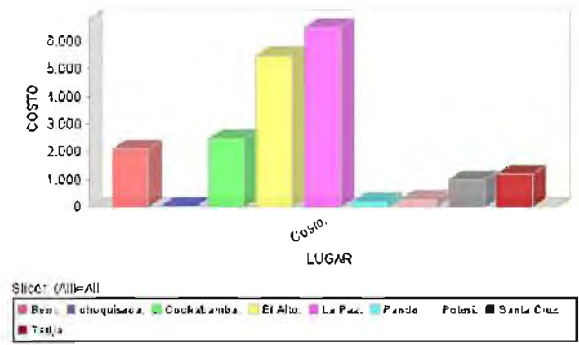
En la figura 3.40 se tiene como resultado el histograma es de la cantidad de las ventas versus el lugar (Oficinas Regionales) , donde se realizaron las ventas, como se puede ver en la figura el lugar donde se realizaron mas ventas es en La Paz, seguido del Beni, el Alto, etc.

Este reporte se puede obtener con el siguiente botón



Información

Slider: [(All)=-All]



Sistema de Mando para SENAMHI

Figura 3.41: Vista con grafico
Fuente: [Elaboración Propia]

En la figura 3.41 se tiene como resultado el histograma es de los costos o ingreso de las ventas versus el lugar (Oficinas Regionales), donde se tuvo más ingresos, como se puede ver en la figura el lugar donde se realizaron mas ventas es en La Paz, seguido el Alto, Cochabamba, etc.

Este reporte se puede obtener con el siguiente botón



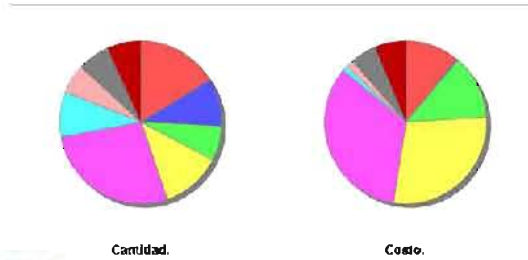
Información

SENAMHI



	Lugar									
Medidas	Beni	Chuquisaca	Cochabamba	El Alto	La Paz	Pando	Potosí	Santa Cruz	Tarija	
Cantidad	51	16	13	23	52	17	11	12	13	
Costo	2.000	50	2.130	5.180	6.180	200	300	1.300	1.200	

Seleccionar: [(All)-All]



Seleccionar: [(All)-All]

- Beni
- Chuquisaca
- Cochabamba
- El Alto
- La Paz
- Pando
- Potosí
- Santa Cruz
- Tarija

Sistema de Mónica - Para SENAMHI

Figura 3.42: Vista con Grafico
Fuente: [Elaboración Propia]

En la figura 3.42 se tiene los resultados en tortas de la cantidad y los costos y el total de las ventas, en cada uno de los lugares (Oficinas Regionales), donde se realizan las ventas, el botón para seleccionar el tipo de grafico que quiere es este

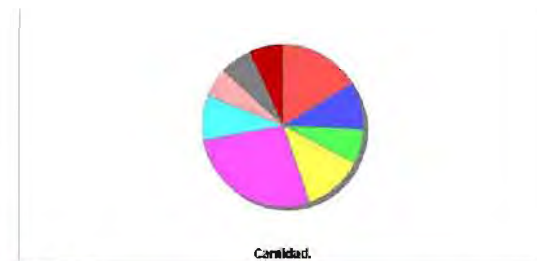


Información

Medidas Cantidad
 Costo
 Ninguno Agrupar Aceptar Cancelar

	Lugar									
Medidas	Beni	Chuquisaca	Cochabamba	El Alto	La Paz	Pando	Potosí	Santa Cruz	Tarija	
Cantidad	1	11	13	23	2	12	1	12	13	

Slicer: [(All) - All]




Slicer: [(All) - All]

Beni
 Chuquisaca
 Cochabamba
 El Alto
 La Paz
 Pando
 Potosí
 Santa Cruz
 Tarija

Sistema de Muestreo Para SENAMHI

Figura 3.43: vista con grafico
Fuente: [Elaboración Propia]

Como podemos en la figura 3.42 estamos realizando la selección de las dimensiones con el botón  en este caso se ha deshabilitado la dimensión de la cantidad solo está habilitado el costo

3.5. Fase de Transición

3.5.1. Casos De Usos Reales

En esta fase, se prioriza los casos de usos reales, que son agrupados en construcciones e iterativos y desarrollados según el orden que evita las vueltas atrás.

En el caso de uso real, el actor percibe cuando interactúa con el sistema y el caso de uso (un caso de uso obtiene valores de resultados observables por un actor en particular).

3.5.2. Interfaces Y Explicaciones De Los Casos De Uso

Los casos de uso reales describen el diseño correcto de casos de uso a partir de una tecnología particular de entrada y salida.

Con la implementación, se muestra los resultados del desarrollo del sistema de acuerdo al diseño elaborado. Esta fase nos lleva a exponer las interfaces graficas.

CASO DE USO REAL DEL SISTEMA



Figura 3.44: Iniciar Sesión
Fuente: [Elaboración Propia]

Caso de uso real: Iniciar Sesión
Actores: Usuario del sistema
Propósito: Validar los datos para ingresar al sistema
Resumen: Realizar en la lista los cambios que requiera.
Tipo: Administrador

Curso normal de eventos	
Acción de los actores	Respuesta del sistema
1. Ingresar sus datos al sistema	2. Sistema verifica datos 3. Datos correctos ingresan al sistema.

Tabla 3.10: Iniciar Sesión
Fuente: [Elaboración Propia]



Figura 3.45: Lista de Clientes
Fuente: [Elaboración Propia]

Caso de uso real: lista clientes
Actores: Usuario del sistema
Propósito: Ver la lista de los clientes
Resumen: Realizar en la lista los cambios que requiera.
Tipo: Administrador

Curso normal de eventos	
Acción de los actores	Respuesta del sistema
<p>1. El sistema por la naturaleza de la información que procesa., tiene establecido la seguridad, que establece para el acceso solo a personal autorizado.</p> <p>2.El usuario ingresa a al listado de los clientes en el cual puede realizar búsquedas.</p> <p>4. El usuario puede adicionar un cliente , modificar datos del cliente y a la vez eliminar al cliente y realizar la proforma que requiera.</p>	<p>3. El Sistema verifica el nombre del cliente y realiza la búsqueda. Y muestra resultados</p>

Tabla 3.11: Lista de clientes
Fuente: [Elaboración Propia]

Curso J2EE - LASIN

Buscador WEB

Traductor de Google

Facebook

SENAMHI

Servicio Nacional de Meteorología e Hidrología
SENAMHI

Estado Plurinacional de Bolivia

ADMINISTRAR PRODUCTOS REPORTES USUARIOS

Opciones

- Cliente
- Regional
- Proforma
- Estacion
- Precios
- volver

Certes

Tipo Cliente:

Razón Social:

Nit / CI:

Dirección:

Teléfono: Fax:

Correo:

Sistema de Maricel Para SENAMHI

Figura 3.46: Nuevo Cliente
Fuente: [Elaboración Propia]

Caso de uso real: Nuevo Cliente	
Actores: Usuario del sistema	
Propósito: Ver detalle de usuarios registrados en el sistema.	
Resumen: El usuario registrar nuevo cliente.	
Tipo: Administrar	
Curso normal de eventos	
Acción de los actores	Respuesta del sistema
1. El evento comienza cuando el usuario elige la opción usuarios nuevo cliente 3. Usuario introduce datos del cliente , selecciona tipo de cliente	2 El Sistema muestra la ventana para realizar nuevo registro de cliente 4. El sistema Almacena los datos Registrados.

Tabla 3.12: Nuevo Cliente
Fuente: [Elaboración Propia]



Figura 3.47: Elaboración de Proforma
Fuente: [Elaboración Propia]

Caso de uso real: Generar Proforma	
Actores: Usuario del sistema	
Propósito: Ver detalle de usuarios registrados en el sistema.	
Resumen: el Usuario Realizara la proforma para el cliente.	
Tipo: Primario esencial	
Curso normal de eventos	
Acción de los actores	Respuesta del sistema
<ol style="list-style-type: none"> 1. El evento comienza cuando el usuario elige la opción Proforma. 2. seleccionar producto 	<ol style="list-style-type: none"> 3. El Sistema devuelve ventana de productos.

Tabla 3.13: Elaboración Proforma
Fuente: [Elaboración Propia]



Figura 3.48: Seleccionar Producto
Fuente: [Elaboración Propia]

Caso de uso real: Seleccionar Producto	
Actores: Usuario del sistema	
Propósito: Ver detalle de usuarios registrados en el sistema.	
Resumen: El usuario Selecciona producto para la proforma.	
Tipo: administrador	
Curso normal de eventos	
Acción de los actores	Respuesta del sistema
<ol style="list-style-type: none"> 1. El evento comienza cuando el usuario elige la opción usuarios del menú principal. 2. El usuario ingresa fecha de inicio y fin del producto que requiere. 3. Seleccionar el Producto que se requiere 4. Ingresa la cantidad de Datos del producto durante el periodo de año que se quiere. 5. finalmente se realiza agregar producto 7. usuario selecciona generar proforma o agrega otro producto. 	<ol style="list-style-type: none"> 6.El Sistema agrega datos a la proforma 8. sistema genera proforma y almacena los datos

Tabla 3.14 Seleccionar producto
Fuente: [Elaboración Propia]



Figura 3.49: Nuevo Producto
Fuente: [Elaboración Propia]

Caso de uso real: Ver usuarios	
Actores: Usuario del sistema	
Propósito: Realizar el Registro de un nuevo producto	
Resumen: El usuario realiza registro de productos	
Tipo: Administrador	
Curso normal de eventos	
Acción de los actores	Respuesta del sistema
1. El sistema por la naturaleza de la información que procesa., tiene establecido la seguridad, que establece para el acceso solo a personal autorizado.	

<p>2.El usuario ingresa a al listado de los producto en el cual puede realizar búsquedas.</p> <p>4. El usuario adicionar seleccionando el tipo de producto y la estación a donde pertenece el producto.</p> <p>6.Usuario puede modificar datos del producto y a la vez eliminar al producto .</p>	<p>3. El Sistema verifica el nombre del producto y realiza la búsqueda. Y muestra resultados</p> <p>5.Sistema almacena datos del producto</p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------

Tabla 3.15: nuevo producto
Fuente: [Elaboración Propia]

The screenshot shows the web interface of the Servicio Nacional de Meteorología e Hidrología (SENAMHI). The header includes the organization's name and logo. Below the header is a navigation menu with options: ADMINISTRAR, PRODUCTOS, REPORTES, and USUARIOS. On the left side, there is a sidebar menu under the heading 'Producto' with sub-items: Cliente, Regional, Profama, Estación, and Precios. The main content area displays a table titled 'Lista de Precios' with the following data:

Precio	Producto	Descripción	Descripción		
0.15	PRECIPITACION	DATO DIARIO POR PARAMETRO	PARAMETRO	<input type="checkbox"/>	<input type="checkbox"/>
30.0	TEMPERATURA	DATO MENSUAL POR PARAMETRO	PARAMETRO	<input type="checkbox"/>	<input type="checkbox"/>
15.0	TEMPERATURA	POR MES	PARAMETRO	<input type="checkbox"/>	<input type="checkbox"/>
600.0	TEMPERATURA	5 AÑOS	PARAMETRO	<input type="checkbox"/>	<input type="checkbox"/>
15.0	VIENTO	DIA	PARAMETRO	<input type="checkbox"/>	<input type="checkbox"/>
50.0	IMAGESE	DIA	SERVICIOS	<input type="checkbox"/>	<input type="checkbox"/>
100.0	LLUVIA	DIA	CERTIFICACION	<input type="checkbox"/>	<input type="checkbox"/>
200.0	PRECIPITACION	MES	PARAMETRO	<input type="checkbox"/>	<input type="checkbox"/>
300.0	IMAGENES SATELITALES	MES	SERVICIOS	<input type="checkbox"/>	<input type="checkbox"/>
100.0	VIENTO	DIA	PARAMETRO	<input type="checkbox"/>	<input type="checkbox"/>

At the bottom of the page, the text 'Sistema de Maricel Para SENAMHI' is visible.

Figura 3.50: Lista de precios
Fuente: [Elaboración Propia]



Figura 3.51: registro de precio
Fuente: [Elaboración Propia]

Caso de uso real: Ver usuarios	
Actores: Usuario del sistema	
Propósito: Realizar el Registro del precio a un determinado producto.	
Resumen: El usuario ve la lista de los precios que se tiene y almacena un precio más	
Tipo: Primario esencial	
Curso normal de eventos	
Acción de los actores	Respuesta del sistema
<ol style="list-style-type: none"> 1. El sistema por la naturaleza de la información que procesa, tiene establecido la seguridad, que establece para el acceso solo a personal autorizado. 2. el usuario selecciona nuevo producto. 3. el usuario selecciona el producto, le añade el precio y la descripción del precio 	<ol style="list-style-type: none"> 4. el sistema almacena los datos del nuevo precio

Tabla 3.16: registro de precio
Fuente: [Elaboración Propia]

**Servicio Nacional de Meteorología e Hidrología
SENAMHI**

ADMINISTRAR PRODUCTOS REPORTES **USUARIO**

Opciones

- Rol de usuario
- usuarios
- Volver

Lista de Usuarios

Buscar: Buscar

Departamento	Regional	Rol de usuario	nombre	apellido	ci			
LA PAZ	LA PAZ	USUARIO	MARCELA	GOMEZ	592558			
SANTA CRUZ	TRINIDAD	INVITADO	MARICEL	ALBERTO	6063663			
PANDO	TRINIDAD	USUARIO	MIGUEL	VEGA	525845			

[Cancelar contraseña](#)

Sistema de Meteo-Hidro-MHI

Figura 3.52: lista de usuario
Fuente: [Elaboración Propia]

Caso de uso real: Ver usuarios	
Actores: Usuario del sistema	
Propósito: Ver detalle de usuarios registrados en el sistema.	
Resumen: Administrador ve lista de usuarios	
Tipo: Primario esencial	
Curso normal de eventos	
Acción de los actores	Respuesta del sistema
<p>1. El sistema por la naturaleza de la información que procesa., tiene establecido la seguridad, que establece para el acceso solo a personal autorizado.</p> <p>2.El usuario ingresa a al listado de los usuarios en el cual puede realizar búsquedas.</p>	<p>3. El Sistema verifica el nombre de los usuarios</p>

4. El usuario puede adicionar un usuarios, modificar datos del cliente y a la vez eliminar al usuarios.	y realiza la búsqueda, y muestra resultados
---------------------------------------------------------------------------------------------------------	---------------------------------------------

Tabla 3.17: lista de usuario
Fuente: [Elaboración Propia]

3.6 Métricas de calidad

3.6.1 Calidad Del SODATD

El control de calidad del SODATD, se realizo mediante cuatro métricas de calidad, que se especifican a continuación:

3.6.2. Primera Prueba

3.6.2.1. Funcionalidad

Para el cálculo de la métrica se hará uso de la medida de punto función, que mide la funcionalidad o utilidad del software.

Primero se determino las cinco características de dominio de información (número de entradas de información, salidas de usuario, peticiones de usuario, archivos e interfaces de usuario), luego se realizo los cálculos de punto función sumando las características del sistema, luego de encontrar los parámetros de medición, el factor de ponderación que se tomo en cuenta fue el factor medio.

PARÁMETROS DE MEDICIÓN	CUENTA		FACTOR DE PONDERACIÓN MEDIO		SUBTOTAL
Número de entradas de usuario	30	*	4	=	120

Número de salidas de usuario	20	*	5	=	100
Número de peticiones de usuario	19	*	4	=	76
Número de archivos	296	*	10	=	2960
Número de interfaces externas	3	*	7	=	21
TOTAL					3277

Tabla 3.18.- Cálculo de puntos de función 1

Fuente.- [Elaboración propia]

Ponderación para el ajuste

0= sin influencia, 1= incidental, 2= moderado, 3= medio, 4= significativo, 5= esencial

No	PREGUNTAS SOBRE LAS CARACTERÍSTICAS DEL SISTEMA	VALORACIÓN
1	¿Requiere el sistema de copias de seguridad y de recuperación?	5
2	¿Se requiere comunicación de datos?	5
3	¿Existen funciones de procesamiento distribuido?	3
4	¿El rendimiento es crítico?	3
5	¿Será ejecutado el sistema en un entorno operativo existente y fuertemente utilizado?	3
6	¿Se requiere entrada de datos en línea?	4
7	¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas o variadas opciones?	3
8	¿Se actualizan los datos maestros de forma interactiva?	4
9	¿Son complejas las entradas, las salidas, los archivos o las peticiones?	3

10	¿Es complejo el procesamiento interno?	4
11	¿Se ha diseñado el código reutilizable?	3
12	¿Están incluidas en el diseño la conversión y la instalación?	4
13	¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	4
14	¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?	4
TOTAL		52

Tabla 3.19.- Preguntas para el cálculo del factor de complejidad 1

Fuente.- [Elaboración propia]

Luego de sumar los puntos asignados a cada respuesta se obtiene un valor F que indica un valor de ajuste de complejidad, luego se reemplazan los valores obtenidos en la ecuación del punto función:

$$PF = 3277 * [0.65 + 0.01 * 52]$$

$$PF = 3834.09$$

El punto función con un ajuste de complejidad máximo es:

$$PF = 3277 * [0.65 + 0.01 * 70]$$

$$PF_{max} = 4423.95$$

Dividimos el punto función hallado entre el punto función de complejidad máxima.

$$\text{Funcionalidad} = PF / PF_{max} = 3834.9 / 4423.95 = 0.86$$

Con lo que se concluye que la funcionalidad es del 86% lo que significa que el SODATD ofrece funciones que satisfacen las necesidades específicas.

3.6.2.2. Fiabilidad

Realizamos los cálculos de “el tiempo medio entre fallas ” y “la disponibilidad mostrada”

MTTF: Tiempo medio hasta la siguiente falla

MTTR: Tiempo medio de reparación

MTBF: Tiempo medio entre fallas

$$MTBF = MTTF + MTTR$$

$$MTBF = 8 \text{ hrs.} + 0.75\text{hrs}$$

$$MTBF = 8.75 \text{ hrs}$$

Ahora se debe calcular la disposición usando la siguiente fórmula:

$$FIABILIDAD = D = [MTTF / (MTTF + MTTR)]$$

$$D = [8 \text{ hrs.} / (8 \text{ hrs} + 0.75 \text{ hrs})]$$

$$D = 0.91$$

Por tanto la Fiabilidad es de 091%

3.6.2.3. Usabilidad

Se debe realizar el test del usuario, una evaluación escrita.

FATOR DE AJUSTE	VALOR
Es entendible	95
Puede ser utilizado fácilmente	90
Es adecuado para mi trabajo	85
Lo Utilizo para determinar tareas	85
Puede ser aprendido con facilidad	95

Puede ser operativo (cambiar configuración)	80
Es Atractivo a la vista	95
Porcentaje total	89%

Por tanto la Usabilidad es de 89%

3.6.2.4 Eficiencia

FATOR DE AJUSTE	VALOR
Tiene un rendimiento de acuerdo a los factores q utiliza	90
Funciona con rapidez cuando realiza funciones	95
Utiliza solo cantidades y recursos necesarios	80
Total	88%

Por tanto la eficiencia es de 88%.

3.6.2.5. Mantenibilidad

Para el cálculo de la facilidad de mantenimiento del sistema, se toma en cuenta algunas preguntas que debe responder el desarrollador

FATOR DE AJUSTE	VALOR
Puede ser modificado el sistema	95
Deja identificar las partes que deben ser modificadas	95
Permite implementar una modificación específica	80
No presenta efectos inesperados en posibles errores	95
Total	91

Por tanto la mantenibilidad es de 91%

3.6.2.6. Portabilidad

El servidor Apache Tomcat 6.0.26, permite su instalación en cualquier sistema operativo Permitiendo la fácil comunicación entre las diferentes terminales de los usuarios. El programa desarrollado SODATD funciona bajo plataforma Windows 2000 profesional y Server, Windows ME, Windows XP.

FATOR DE AJUSTE	VALOR
Puede ser transferido de un entorno a otro	95
es capaz de ser adaptado a otros ambientes con facilidad.	70
Es fácil de instalar	95
Total	86.6

Por tanto la portabilidad es de 87 %

3.6.2.7 Confiabilidad

Recordando que la confiabilidad representa el tiempo que el software esta disponible para su uso, se calculará la probabilidad de ejecución del sistema libre de fallos en una gestión, para lo cual se hará uso de la función de confiabilidad dada por:

$$F(t) = f * e^{-\lambda/10 * t}$$

La funcionalidad del sistema es del 86% y t representa el tiempo que dura una gestión en la institución, lambda es la probabilidad de error que pueda tener el sistema, donde diez es el número de ejecuciones del sistema durante un mes. Reemplazando datos tenemos:

$$F(t) = 0.86 * e^{-1/10 * 12}$$

$$0.86 * e^{-1/10 * 12} = 0.25$$

Por lo tanto la probabilidad de que existan fallas en el sistema es del 25%.

La probabilidad de operación del sistema libre de fallos en una gestión será calculada con las siguientes expresiones.

$$F(t) = 1-[0.86 * e^{(-1/10*t)}]$$

$$F(t) = 1-[0.86 * e^{(-1/10*12)}]$$

Realizando las operaciones respectivas, obtenemos:

$$1-[0.86 * e^{(-1/10*12)}] = 0.75$$

La confiabilidad o inexistencia de fallas en el sistema durante una gestión es del 75%

3.6.3. Segunda Prueba

3.6.3.1 Funcionalidad

PARÁMETROS DE MEDICIÓN	CUENTA		FACTOR DE PONDERACIÓN MEDIO		SUBTOTAL
Número de entradas de usuario	50	*	4	=	200
Número de salidas de usuario	35	*	5	=	525
Número de peticiones de usuario	19	*	4	=	76
Número de archivos	450	*	10	=	450
Número de interfaces externas	3	*	7	=	21
TOTAL					1272

Tabla 3.20.- Cálculo de puntos de función 2

Fuente.- [Elaboración propia]

Ponderación para el ajuste

0= sin influencia, 1= incidental, 2= moderado, 3= medio, 4= significativo, 5= esencial

No	PREGUNTAS SOBRE LAS CARACTERÍSTICAS DEL SISTEMA	VALORACIÓN
1	¿Requiere el sistema de copias de seguridad y de recuperación?	5
2	¿Se requiere comunicación de datos?	5
3	¿Existen funciones de procesamiento distribuido?	3
4	¿El rendimiento es crítico?	3
5	¿Será ejecutado el sistema en un entorno operativo existente y fuertemente utilizado?	3
6	¿Se requiere entrada de datos en línea?	4
7	¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas o variadas opciones?	4
8	¿Se actualizan los datos maestros de forma interactiva?	5
9	¿Son complejas las entradas, las salidas, los archivos o las peticiones?	4
10	¿Es complejo el procesamiento interno?	4
11	¿Se ha diseñado el código reutilizable?	3
12	¿Están incluidas en el diseño la conversión y la instalación?	4
13	¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	4
14	¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?	4
TOTAL		55

Tabla 3.21.- Preguntas para el cálculo del factor de complejidad 2

Fuente.- [Elaboración propia]

Luego de sumar los puntos asignados a cada respuesta se obtiene un valor F que indica un valor de ajuste de complejidad, luego se reemplazan los valores obtenidos en la ecuación del punto función:

$$PF = 1272*[0.65+0.01*55]$$

$$PF = 1526.4$$

El punto función con un ajuste de complejidad máximo es:

$$PF = 1272*[0.65+0.01*70]$$

$$PF_{max} = 1717.2$$

Dividimos el punto función hallado entre el punto función de complejidad máxima.

$$\text{Funcionalidad} = PF/PF_{max} = 1526.4/1717.2 = 0.88$$

Con lo que se concluye que la funcionalidad es del 88%

3.6.3.2. Fiabilidad

Realizamos los cálculos de “el tiempo medio entre fallas” y “la disponibilidad mostrada”

MTTF: Tiempo medio hasta la siguiente falla

MTTR: Tiempo medio de reparación

MTBF: Tiempo medio entre fallas

$$**MTBF = MTTF + MTTR**$$

$$MTBF = 9 \text{ hrs.} + 0.75\text{hrs}$$

$$MTBF = 9.75 \text{ hrs}$$

Ahora se debe calcular la disposición usando la siguiente fórmula:

$$FIABILIDAD = D = [MTTF / (MTTF + MTTR)]$$

$$D = [9 \text{ hrs.} / (9 \text{ hrs} + 0.75 \text{ hrs})]$$

$$D = 0.92$$

Por tanto la Fiabilidad es de 092%

3.6.3.3. Usabilidad

Se debe realizar el test del usuario, una evaluación escrita.

FATOR DE AJUSTE	VALOR
Es entendible	95
Puede ser utilizado fácilmente	96
Es adecuado para mi trabajo	90
Lo Utilizo para determinar tareas	85
Puede ser aprendido con facilidad	95
Puede ser operativo (cambiar configuración)	80
Es Atractivo a la vista	95
Porcentaje total	90%

Por tanto la Usabilidad es de 90%

3.6.3.4. Eficiencia

FATOR DE AJUSTE	VALOR
Tiene un rendimiento de acuerdo a los factores q utiliza	92
Funciona con rapidez cuando realiza funciones	95
Utiliza solo cantidades y recursos necesarios	80
Total	89%

Por tanto la eficiencia es de 89%.

3.6.3.5. Mantenibilidad

Para el cálculo de la facilidad de mantenimiento del sistema, se toma en cuenta algunas preguntas que debe responder el desarrollador

FATOR DE AJUSTE	VALOR
Puede ser modificado el sistema	95
Deja identificar las partes que deben ser modificadas	95
Permite implementar una modificación específica	80
No presenta efectos inesperados en posibles errores	95
Total	91

Por tanto la mantenibilidad es de 91%

3.6.3.6. Portabilidad

El servidor Apache Tomcat 6.0.26, permite su instalación en cualquier sistema operativo Permitiendo la fácil comunicación entre las diferentes terminales de los usuarios. El programa desarrollado SODATD funciona bajo plataforma Windows 2000 profesional y Server, Windows ME, Windows XP.

FATOR DE AJUSTE	VALOR
Puede ser transferido de un entorno a otro	95
es capaz de ser adaptado a otros ambientes con facilidad.	70
Es fácil de instalar	95
Total	86.6

Por tanto la portabilidad es de 87 %

3.6.3.7. Confiabilidad

Recordando que la confiabilidad representa el tiempo que el software esta disponible para su uso, se calculará la probabilidad de ejecución del sistema libre de fallos en una gestión, para lo cual se hará uso de la función de confiabilidad dada por:

$$F(t) = f * e^{-\lambda/10 * t}$$

La funcionalidad del sistema es del 86% y t representa el tiempo que dura una gestión en la institución, lambda es la probabilidad de error que pueda tener el sistema, donde diez es el número de ejecuciones del sistema durante un mes. Reemplazando datos tenemos:

$$F(t) = 0.88 * e^{-1/10 * 12}$$

$$0.88 * e^{-1/10 * 12} = 0.26$$

Por lo tanto la probabilidad de que existan fallas en el sistema es del 26%.

La probabilidad de operación del sistema libre de fallos en una gestión será calculada con las siguientes expresiones.

$$F(t) = 1 - [0.88 * e^{-1/10 * t}]$$

$$F(t) = 1 - [0.88 * e^{-1/10 * 12}]$$

Realizando las operaciones respectivas, obtenemos:

$$1 - [0.88 * e^{-1/10 * 12}] = 0.74$$

La confiabilidad o inexistencia de fallas en el sistema durante una gestión es del 74%

3.7. Tabla Comparativa de Pruebas

	PRIMERA PRUEBA	SEGUNDA PRUEBA
Funcionalidad	86	88
Fiabilidad	91	92
Usabilidad	89	90
Eficiencia	88	89
Mantenibilidad	91	91

Portabilidad	87	87
Confiabilidad	75	74

Tabla 3.22.- Tabla Comparativa

Fuente.- [Elaboración propia]

3.8. Costos y Beneficios

Existen técnicas para la estimación de costos, pero para ello se requiere experiencia.

El manejador de costo principal para un proyecto de desarrollo de software es sin duda el tamaño del producto. La medida del tamaño debe ser tal que este en relación directa con el esfuerzo de desarrollo, por lo que las métricas de tamaño tratan de considerar todos los aspectos que influyen en el costo.

3.8.1 Análisis De Costo.

El modelo constructivo de costes o COCOMO, es un modelo de estimación de costes de software, orientado a la magnitud del producto final, midiendo el tamaño del proyecto en líneas de código principalmente. COCOMO ayuda a estimar el esfuerzo, tiempo, gente y costos. El modelo provee tres niveles de aplicación: básico, intermedio y avanzado, basados en los factores considerados por el modelo.

Básico, es un modelo estático simplemente evaluado que calcula el esfuerzo (y costo) del desarrollo del software como función del programa expresado en líneas de código. Intermedio, calcula el esfuerzo del desarrollo del software como función del tamaño del programa y un conjunto de guías de costo que incluye una evaluación subjetiva del producto, hardware, personal y de los atributos del

proyecto. Avanzado, incorpora todas las características de la versión intermedia con una evaluación del impacto de las vías de costo en cada fase del proceso de la ingeniería de software.

En cada nivel de aplicación están definidos para tres tipos de proyectos de software: modo orgánico, proyectos de software relativamente pequeños y sencillos en los que pequeños equipos con buena experiencia en la aplicación trabajan en un conjunto de requerimiento poco rígido. Modo semi-acoplado, un

proyecto de software intermedio en tamaño u complejidad en el cual equipos con distintos niveles de experiencia debe satisfacer requerimientos poco y medio rígidos. Modo acoplado, un proyecto de software que debe ser desarrollado dentro un conjunto estricto de hardware, software y de restricciones operativas.

La ecuación de COCOMO en este modo básico es:

$$\begin{aligned} E &= a \cdot KLOC^b \\ D &= c \cdot E^d \\ P &= E/D \end{aligned}$$

Donde:

E: es el esfuerzo aplicado en persona por mes,

D: es el tiempo de desarrollo en meses

KLOC: es el número de líneas estimadas para el proyecto (en miles)

P: es el número de personas necesarias.

Los coeficientes **a**, **b**, **c**, y **d** se obtienen de la siguiente tabla:

MODE DEDESARROLLO

	A	B	C	d
Organico	2.4	1.05	2.5	0.38
semiacoplado	3.0	1.12	2.5	0.35
acoplado	3.6	1.20	2.5	0.32

Tabla 3.23.- Tabla de los Coeficientes para los Diferentes Proyectos

Fuente [elaboración propia]

Para el proyecto se realizaron los siguientes cálculos considerando el modo semiacoplado:

$$a = 3.0$$

$$b = 1.12$$

$$c = 2.5$$

$$d = 0.35$$

$$KLOC = 6$$

Reemplazando

$$E = a \cdot KLOC^b = 3 \cdot 6^{1.12}$$

$$E = 25.46 \text{ PERSONA - MES}$$

Hallando el tiempo de desarrollo

$$D = c \cdot E^d = D = 2.5 \cdot 25.46^{0.35}$$

$$D = 7.76 \text{ mes}$$

Número de personas necesarias

$$P = E / D = 25.46 / 7.76$$

$$P = 3.28 \text{ Personas.}$$

Considerando que el sueldo del desarrollador dependen de la experiencia del mismo y es un valor muy subjetivo, se da un valor según la oferta de los programadores en el mercado de 2500 Bs.

Los costos realizados se muestran en la tabla siguiente.

DESCRIPCION CANTIDAD COSTO

	DESCRIPCION CANTIDAD COSTO	MEN SUAL	CANTIDAD DE MESES	TOTAL
Desarrolladores	3	2500	7	25500
Equipos	6	-	Ya existe	-
Software	Libre	-		-
Adiestramiento	-	1000	1/2	500
Otros	-	500	-	500

Tabla3.24. Tablas de los Costos del Sistema
Fuente [elaboración propia]

Lo que lleva a que el costo total sea de 52500 Bs.

3.8.2. Beneficios

El sistema proveerá acceso y transferencia de información en tiempo real, entre los diferentes usuarios, de tal forma que la información es oportuna en el momento necesario para los mismos. Los beneficios tangibles que se pueden mencionar son:

- a) todos los datos estarán centralizados en una sola base de datos.
- b) La información se transfiere electrónicamente a diferentes usuarios según los Privilegios que tenga.

- c) Se puede hacer informes con la información en tiempo real
- d) Se evita el gasto innecesario de papel reduciendo gastos operacionales.
- e) Con el software se reduce el tiempo en la productividad del personal, redundando.



4.1. Conclusiones

Al concluir el presente proyecto se observa que se ha realizado todos los objetivos propuesto al inicio y se llego a las siguientes conclusiones:

El sistema desarrollado es capaz de reemplazar el proceso manual de la institución ya que cumple con todo el proceso de venta de certificaciones de datos meteorológicos e hidrológicos del SENAMHI(Servicio Nacional de Meteorología e Hidrología)

- ✓ Cuenta con el diseño de base de datos relacional el cual hace uso de los servicios OLAP en base al servidor Mondrian.
- ✓ Realiza el control de rol de usuarios para la administración del sistema
- ✓ Realizara el control de los registros de los clientes que ha realizado solicitudes de proformas a la institución.
- ✓ Realiza el control de la Proformas elaboradas de los productos que la institución ofrece.
- ✓ Además realiza el control de los Productos que la institución tiene para facilitar al cliente
- ✓ El control de las Estaciones las cuales son los puntos donde se tienen los parámetros para realizar las proformas
- ✓ El control de la oficinas regionales donde se realizan la venta de certificaciones
- ✓ Realizar el control de las ventas de la institución en base a al tiempo que el usuario requiera. En base a histogramas.

4.2. Recomendaciones

Es de suma importancia enfatizar el diseño del proyecto realizado las siguientes observaciones:

Realizar la supervisión del sistema cada mes para el buen funcionamiento del mismo en el proceso de almacenamiento de información y la emisión de los reportes.

Realizar copias de seguridad diariamente para evitar la pérdida de información

Se recomienda realizar posteriormente un sistema que obtenga la Imágenes satelitales y así obtener el pronóstico del tiempo para la institución.

Un Sistema para realizar el control de inventarios que permita realizar administrar los datos existentes y los datos procesados.



4.3 Bibliografía

[ConsolidaIT, 2008], ConsolidaIT.(2008) Fases RUP, Disponible en: <http://consolida-it.com/marco.htm>

[Crawford, 2003] Crawford William and Kaplan Jonathan, J2EE DesignPatterns, edición, Septiembre de 2003, Editorial: RA-MA.

[Cay, 2007], Cay Geary David Horstmann Core Java Server Faces (2007), edición, Segunda edición Abril de 2007, Editorial Ra-Ma

[Evitts, 2000], Evitts Paul. A UML Pattern Language, edición, Indianapolis 2000, Editorial, Macmillan Technical Publishing

[Inmon, 2010], Inmon Bill. (2010) DataWarehouse, Disponible en: http://www.sinnexus.com/business_intelligence/datawarehouse.aspx

[Jacobson, Booch, Rumbaugh, 2000.], Jacobson, Ivar; Booch, Grady; Rumbaugh, James. El Proceso Unificado de Desarrollo de Software, edición, Pearson Addison 2000, Editorial: Wesley.

[Kobayashi, 2007], Kobayashi G. La industria del Software en el Peru, Disponible en :<http://www.coeris.org>

[Microsoft , 2000], Microsoft Books Online (2000). Cubo de datos, Disponible en: <http://www.monografias.com/trabajos57/data-warehouse-sql/data-warehouse-sql2.shtml>

[Pentaho, 2007], Pentaho (2007) Funcionamiento Mondrian, Disponible en: <http://mondrian.pentaho.com/>

[Postgres, 2010] Motor de bases de datos PostgreSQL (2010), Disponible en: <http://www.postgresql.org/docs/>

[Scalet, 2000], Scalet (2000) Aspectos de la integración de ISO/IEC 9126,
Disponible en: http://www.worldlingo.com/ma/enwiki/es/ISO_9126

[Wikipedia, 2008], Wikipedia. (2008) Representación Esquema Star, Disponible en:
http://es.wikipedia.org/wiki/Esquema_Star

[Wikipedia, 2010], Wikipedia (2010) Representación de Tabla de Hechos,
Disponible en: http://es.wikipedia.org/wiki/Tabla_de_hechos

