UNIVERSIDAD NACIONAL DE LOJA

ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO RENOVABLES

CARRERA DE INGENIERÍA EN SISTEMAS

TEMA:

"DISEÑO Y CREACIÓN DEL MÓDULO DE PRONÓSTICO DE CARGA, PARA EL DESARROLLO DE UN SOFTWARE PARA EL ESTUDIO DE REDES DE DISTRIBUCIÓN ELÉCTRICA DE LA CIUDAD DE LOJA".

PROYECTO DE TESIS PREVIA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS

AUTOR:

Soraya Yolanda Carrión González

ASESOR: Ing. Franco Hernán Salcedo López, Mg. Sc.

Loja – Ecuador

2014

CERTIFICACIÓN

Ing. Franco Hernán Salcedo López, Mg. Sc.
DIRECTOR DEL PROYECTO DE TESIS

Haber dirigido, asesorado, revisado y corregido el presente trabajo de tesis de grado, en su proceso de investigación, bajo el tema **"DISEÑO Y CREACIÓN DEL MÓDULO DE PRONÓSTICO DE CARGA, PARA EL DESARROLLO DE UN SOFTWARE PARA EL ESTUDIO DE REDES DE DISTRIBUCIÓN ELÉCTRICA DE LA CIUDAD DE LOJA"**, previa a la obtención del título de **Ingeniero en Sistemas**, realizado por la Egresada **Soraya Yolanda Carrión González**, la misma que cumple con la reglamentación y políticas de investigación, por lo que autorizo su presentación y posterior sustentación y defensa.

Loja, junio de 2014.

Ing. Franco Hernán Salcedo López, Mg. Sc.
DIRECTOR DE TESIS

Autoría

Yo, **Soraya Yolanda Carrión González** declaro ser autora del presente trabajo de tesis y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la publicación de mi tesis en el repositorio institucional-Biblioteca Virtual.

Autor: Soraya Yolanda Carrión González

Firma:…… ……..

Cedula: 1104612765

Fecha: 2 de julio del 2014

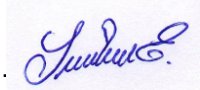
Carta De Autorización De Tesis Por Parte Del Autor, Para La Consulta, Reproducción Parcial O Total, Y Publicación Electrónica Del Texto Completo.

Yo **Soraya Yolanda Carrión González**, declaro ser autor de la tesis titulada: **"DISEÑO Y CREACIÓN DEL MODULO DE PRONÓSTICO DE CARGA, PARA EL DESARROLLO DE UN SOFTWARE PARA EL ESTUDIO DE REDES DE DISTRIBUCIÓN ELÉCTRICA DE LA CIUDAD DE LOJA"**., como requisito para optar al grado de: **Ingeniero en Sistemas**; autorizo al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Digital Institucional:

Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja, a los dos días del mes de julio del dos mil catorce, firma del autor.

Firma: ……  ……
Autor: Soraya Yolanda Carrión González
Cédula: 1104612765
Dirección: San José Bajo **(**Carlos Román 04-17 y Bolívar Bailón**)**
Correo electrónico: flakis_1489@hotmail.com
Teléfono: 072561598 **Celular:** 0994643410

DATOS COMPLEMENTARIOS
Director de Tesis: Ing. Franco Hernán Salcedo López, Mg. Sc.
Tribunal de Grado: Ing. Marco Augusto Ocampo Carpio, Mg. Sc.
Ing. Henry Francisco Cueva Bravo, Mg. Sc.
Ing. Henry Patricio Paz Arias, Mg. Sc.

Agradecimiento

A la Universidad Nacional de Loja, porque en sus aulas, recibí el conocimiento intelectual y humano de cada uno de los docentes de la Carrera de Ingeniería en Sistemas.

Especial agradecimiento a mi Director de Tesis el Ing. Franco Salcedo y al Asesor del Proyecto Ing. Jorge Carrión por su ayuda desinteresada en el transcurso del desarrollo del mismo, por sus consejos y amistad.

La Autora

Dedicatoria

A Dios, verdadera fuente de amor y sabiduría.

A mi padre Alonso, porque gracias a él sé que la responsabilidad se la debe vivir como un compromiso de dedicación y esfuerzo.

A mi madre Mariela, cuyo vivir me ha mostrado que en el camino hacia la meta se necesita de la dulce fortaleza para aceptar las derrotas y del sutil coraje para derribar miedos.

A mis hermanos Jorge y Marianela, el incondicional abrazo que me motiva y recuerda que detrás de cada detalle existe el suficiente alivio para empezar nuevas búsquedas.

A mis familiares, viejos amigos y a quienes recién se sumaron a mi vida para hacerme compañía con sus sonrisas de ánimo, en especial a mi esposo Israel y a mi hijo Adriancito ya que ellos son mi impulso y fortaleza para seguir adelante.

Soraya

a. Título

“DISEÑO Y CREACIÓN DEL MÓDULO DE PRONÓSTICO DE CARGA, PARA EL DESARROLLO DE UN SOFTWARE PARA EL ESTUDIO DE REDES DE DISTRIBUCIÓN ELÉCTRICA DE LA CIUDAD DE LOJA”.

b. Resumen

El presente proyecto de tesis se lo desarrollo con el propósito de otorgar a la EERSSA un módulo de pronóstico de demanda, como un aporte a la creación de un software para el estudio técnico económico del régimen de operación de las redes de distribución eléctrica de la ciudad de Loja, el mismo que permitirá realizar estudios en las redes, en particular el pronóstico de la demanda de energía eléctrica, ya es una necesidad básica para conocer el estado de las instalaciones eléctricas en el tiempo y además sirve de base de cálculo en todos los estudios de las redes de distribución, es utilizado continuamente por las diferentes empresas dedicadas a la distribución de energía eléctrica, así como también servirá de aporte para la formación al más alto nivel en redes de distribución de los estudiantes de pregrado y postgrado del área de Energía.

Es por ello que se desarrolló el módulo de escritorio que permita el procesamiento de datos para el pronóstico de la demanda de energía eléctrica, que a través de métodos de regresión (mínimos cuadrados), correlación y pronósticos realiza el cálculo del pronóstico de demanda en el intervalo de año-mes-día, y el cálculo de la varianza, la pendiente, T de Student, la desviación estándar, el nivel de certidumbre, el nivel de incertidumbre y graficar los años correlacionados.

El método de pronóstico selecciona los valores por trimestres, de los últimos 5 años anteriores y así determina un valor aproximado de consumo para el siguiente año.

Para realizar los cálculos se procede a escoger los datos correspondientes ya sea para usar los valores de P (Potencia Activa) o Q (Potencia Reactiva) y así dar el tratamiento a la información seleccionada y se procede a definir el tipo de cálculo ya sea por año mes o día, además de seleccionar la subestación y el alimentador a calcular.

Se realizó el análisis y diseño preliminar del software: describir los casos de uso, diagramas de robustez y diagrama de clases, después se procedió a programar los métodos y modelos para el pronóstico de carga y se diseñó la interfaz gráfica de usuario en base a los métodos y modelos del sistema.

Se procedió a realizar las pruebas necesarias para determinar que el software está acorde a los requerimientos de la empresa con el Ing. Jorge Carrión el mismo que es encargado del proyecto en mención por parte de la UNL y la EERSSA.

En el desarrollo del módulo del proyecto de tesis se trabajó con la metodología ICONIX y el patrón MVC (Modelo-Vista-Controlador), así mismo se usó tecnologías de distribución libre JAVA con su entorno de programación IDE NetBeans 8.0, y el gestor de base de datos MySQL 6.1.

Se concluye que el programa cumple con los objetivos planteados y satisface las necesidades de la EERSSA.

Se recomienda hacer uso de la aplicación ya que tiene una interfaz amigable con el usuario y determina los cálculos en un tiempo mínimo comparado con el tiempo que llevaba hacer los cálculos manuales.

Summary

The present draft of the thesis is the development with the purpose to give the module a EERSSA forecast demand, as a contribution to the creation of a software for the technical study of the economic regime of operation of the electricity distribution networks in the city of Loja, The same that will allow you to perform studies on the networks, in particular the forecast of the demand for electric power, because it is a basic need to know the state of electrical installations in the time and also serves as the basis for calculation in all the studies of the distribution networks.

The module enables the processing of data for the forecast of the demand for electric power, that through regression methods (least squares), correlation forecasts and performs the calculation of the prognosis of demand in the range of year-month-day, and the calculation of the variance, the slope, Student's t test, the standard deviation, the level of certainty, the level of uncertainty and plotting the years correlated.

The analysis is performed and preliminary design of the software: describe use cases, diagrams of robustness and class diagram, then proceeded to program the methods and models for the prediction of load and design the graphical user interface based on the methods and models of the system.

It is recommended that you make use of the application as it has a friendly interface with the user and determines the calculations in a minimum time compared with the time it took to do manual calculations.

INDICE DE CONTENIDOS

Certificación del Director II

Autoría III

Carta De Autorización De Tesis Por Parte Del Autor, Para La Consulta, Reproducción Parcial O Total, Y Publicación Electrónica Del Texto Completo. III

Agradecimiento IVV

Dedicatoria VI

a. Título 7

b. Resumen 8

Summary 10

ÍNDICE DE CONTENIDOS 12

ÍNDICE DE FIGURAS 16

ÍNDICE DE TABLAS 18

c. INTRODUCCIÓN 19

d. REVISIÓN LITERARIA 23

1. UML 23

1.1. Modelado De Objetos 24

1.2. Artefactos Para El Desarrollo De Proyectos 27

1.2.1. Diagramas de Implementación 28

1.2.2. Diagramas De Componentes 28

1.2.3. Diagrama De Plataformas O Despliegue 28

1.2.4. Diagramas De Interacción O Comportamiento 29

1.2.5. Diagrama de secuencia 29

1.2.6. Diagrama de colaboración 30

1.2.7. Diagramas De Actividad 31

1.2.8. Diagramas De Estado 31

1.2.9. Diagramas De Casos De Uso 32

1.2.10. Diagramas De Clases 33

1.2.11. Relación De Refinamiento 36

1.2.12. El Proceso De Desarrollo 37

2. Metodología Iconix 37

2.1. Análisis De Requerimientos 39

2.2. Análisis y Diseño Preliminar 51

2.3. Resumen ICONIX 56

3. Antecedentes 58

3.1. Redes Eléctricas 59

3.5.5. Subestación de Distribución 69

e. MÉTODOS Y TÉCNICAS 75

1. MÉTODOS 75

1.1. Método Deductivo 75

1.2. Método Inductivo 75

2. METODOLOGÍA DE DESARROLLO 76

2.1. ESTA METODOLOGÍA ESTÁ COMPUESTA POR CUATRO FASES PRINCIPALES: 76

2.1.1. Análisis de Requisitos 76

2.1.2. Análisis y Diseño preliminar 76

2.1.3. Diseño 77

2.1.4. Implementación y Pruebas 77

2.2. Ventajas de Iconix 77

f. RESULTADOS 78

1. ANÁLISIS DE REQUERIMIENTOS 78

ANÁLISIS DE REQUISITOS 78

RECOLECCIÓN DE LA INFORMACIÓN 79

REQUERIMIENTOS FUNCIONALES 79

REQUERIMIENTOS FUNCIONALES 79

MODELO DEL DOMINIO 81

MODELO DE CASOS DE USO 82

2. Análisis Y Diseño Preliminar 85

2.1. Descripción De Casos De Uso 85

PROTOTIPO INICIAL 86

3. DISEÑO 101

3.1. DIAGRAMAS DE SECUENCIA 101

3.2. DIAGRAMAS DE ROBUSTEZ 109

3.3. Diagrama de clases 113

3.4. Diagrama De Base De Datos 114

4. IMPLEMENTACIÓN 117

4.1 Método de Operaciones Base de Datos 117

4.2. Método de Regresión 118

4.3. Método de Correlación 119

4.4. Método de Pronóstico 122

4.5. Cálculo Grados de Libertad y Rango de Certidumbre 123

g. DISCUSIÓN 129

1. Desarrollo De La Propuesta Alternativa 129

h. CONCLUSIONES 136

i. RECOMENDACIONES 138

j. Bibliografía 139

ANEXOS ..... **¡Error! Marcador no definido.**

Anexo 1. Método de regresión (mínimos cuadrados) ..... **¡Error! Marcador no definido.**

ÍNDICE DE FIGURAS

FIGURA 1. FLUJOS DE TRABAJO DE ICONIX.....38
FIGURA 2. . MODELO DE DOMINIO, EN CONTEXTO DEL PROCESO ICONIX 40

FIGURA 3. NOTACIÓN PARA REPRESENTAR UNA CLASE....42
FIGURA 4. RELACIONES PARA ASOCIAR LOS OBJETOS DE DOMINIO....43
FIGURA 5. EJEMPLO DE UN MODELO DE DOMINIO....43
FIGURA 6. MODELO DE CASOS DE USO, EN CONTEXTO DEL PROCESO ICONIX....44
FIGURA 7. REPRESENTACIÓN GRÁFICA DE UN ACTOR....45
FIGURA 8. RELACIONES PARA ASOCIAR LOS CASOS DE USO....47
FIGURA 9. EJEMPLO DE UN DIAGRAMA DE CASOS DE USO....48
FIGURA 10. ESCENARIO DE UN CASO DE USO....50
FIGURA 11. ANÁLISIS DE ROBUSTEZ, EN CONTEXTO DEL PROCESO ICONIX....52
FIGURA 12. SÍMBOLOS UTILIZADOS EN EL DIAGRAMA DE ROBUSTEZ....53
FIGURA 13. SISTEMA DE SUMINISTRO ELÉCTRICO....64
FIGURA 14. TRANSFORMADOR DE DISTRIBUCIÓN....72
FIGURA 15. ESQUEMA ELÉCTRICO, TRANSFORMADOR MONOFÁSICO....73
FIGURA 16. CONDUCTOR DE ALUMINIO XLPE AISLADO Y PVC ENFUNDADOS CABLE ELÉCTRICO....74
FIGURA 17. MODELO DEL DOMINIO....81
FIGURA 18. CASO DE USO GENERAL....85
FIGURA 19. PANTALLA PRINCIPAL....86
FIGURA 20. PRONÓSTICO DE CARGA....88
FIGURA 21. MÉTODO DE REGRESIÓN....88
FIGURA 22. MÉTODO CORRELACIÓN....91
FIGURA 23. MÉTODO PRONÓSTICO....91
FIGURA 24. DIAGRAMA DE SECUENCIA PRONÓSTICO DE CARGA....102
FIGURA 25. DIAGRAMA DE SECUENCIA PRONÓSTICO POR MÉTODO REGRESIÓN....104
FIGURA 26.DIAGRAMA DE SECUENCIA PRONÓSTICO POR MÉTODO CORRELACIÓN....107
FIGURA 27. DIAGRAMA DE SECUENCIA MÉTODO POR PRONÓSTICO....109
FIGURA 28. DIAGRAMA DE ROBUSTEZ PRONÓSTICO DE CARGA....109
FIGURA 29. DIAGRAMA DE ROBUSTEZ PRONÓSTICO POR MÉTODO DE REGRESIÓN....110
FIGURA 30. DIAGRAMA DE ROBUSTEZ PRONÓSTICO DE CORRELACIÓN....111

FIGURA 31. DIAGRAMA DE ROBUSTEZ PRONÓSTICO POR MÉTODO DE PRONÓSTICO 112
FIGURA 32. DIAGRAMA DE CLASES 113
FIGURA 33. DIAGRAMA DE BASE DE DATOS 114

ÍNDICE DE TABLAS

TABLA I RESUMEN DE ICONIX 57
TABLA II. REQUERIMIENTOS FUNCIONALES 79
TABLA III. REQUERIMIENTOS NO FUNCIONALES 80
TABLA IV. DESCRIPCIÓN MODELO DEL DOMINIO 81
TABLA V. PRONÓSTICO DE CARGA 92
TABLA VI. CALCULAR MÉTODO DE REGRESIÓN 94
TABLA VII. CALCULAR MÉTODO DE CORRELACIÓN 96
TABLA VIII. PRONOSTICAR POR AÑO SIGUIENTE 98
TABLA IX. RECURSOS HUMANOS 132
TABLA X. RECURSOS MATERIALES 133
TABLA XI. RECURSOS DE HARDWARE 133
TABLA XII. RECURSOS DE SOFTWARE 134
TABLA XIII. RECURSOS DE COMUNICACIÓN 135
TABLA XIV. RESUMEN DEL PRESUPUESTO 135

c. INTRODUCCION

La Universidad Nacional de Loja, a través de su departamento de Investigación y desarrollo crea el proyecto “DISEÑO Y CREACIÓN DE UN SOFTWARE PARA EL ESTUDIO TÉCNICO ECONÓMICO DEL RÉGIMEN DE OPERACIÓN DE LAS REDES DE DISTRIBUCIÓN ELÉCTRICA DE LA CIUDAD DE LOJA.”, ya que ve las necesidades que tiene la EERSSA y esta es la oportunidad para que se actualice y emprenda nuevos proyectos y abarque todos los puntos en donde aún no se ha podido llegar con la energía eléctrica dejando a un lado la brecha digital que divide a un pueblo.

En la actualidad la Universidad Nacional de Loja (U.N.L) mantiene un convenio de trabajo con la Empresa Eléctrica Regional del Sur SA (EERSSA), la cual posee una capacidad instalada en subestaciones superior a los 40 MW para brindar servicio eléctrico a la ciudad de Loja. Cuenta con 18 circuitos de distribución para abastecer de energía a toda la ciudad de Loja. Existen en el territorio diversas instalaciones del sector de la mediana y pequeña industria que requieren especial atención en lo referente al uso racional de la energía.

La ciudad de Loja ha tenido un crecimiento poblacional apreciable en los últimos años, y con ello un aumento en la demanda de las redes de distribución de la energía eléctrica; sin haberse realizado un estudio integral del sistema de redes como un todo que cumpla con las normas y reglamentos que estipulan los entes reguladores en el Ecuador; es decir, el Consejo Nacional de Electricidad (CONELEC) y el Centro Nacional de Control de Energía (CENACE), es por ello que existe la necesidad de desarrollar métodos de predicción de la demanda eléctrica que permitan proyectar la magnitud de las cargas futuras, con vista a definir las medidas para cumplir las normas y reglamentos estipulados, a través del análisis realizado se determina la falta de sistema para pronosticar con precisión el consumo de energía eléctrica a corto, mediano y largo plazo y la falta de algoritmos para el estudio

técnico-económico de la evaluación de los parámetros de operación y funcionamiento de los circuitos.

Se determinó como objetivos el definir el lenguaje de programación que este afín a las necesidades del proyecto, trabajar con las normativas para la evaluación de redes eléctricas estipuladas por la EERSSA y el CONELEC, determinar los requisitos funcionales y no funcionales de la empresa, desarrollar un diagrama de clases de alto nivel definido como modelo de dominio, determinar los casos de uso del software, realizar el análisis y diseño preliminar del software: describir los casos de uso, diagramas de robustez y diagrama de clases, programar los métodos y modelos para el pronóstico de carga, diseñar la interfaz gráfica de usuario en base a los métodos y modelos del sistema y realizar las pruebas necesarias para determinar que el software está acorde a los requerimientos de la EERSSA.

Para el desarrollo del presente proyecto se utilizó una metodología que establezca una secuencia de pasos ordenados, los cuales sirvieron de guía a lo largo de la ejecución de diferentes actividades, alcanzando de esta manera los resultados que se esperaban.

Se aplicó el método deductivo el cual permitió conocer de manera específica cada uno de los problemas más comunes y que se presentan generalmente en las redes de distribución de energía eléctrica de la EERSSA, en particular el crecimiento de la demanda de energía eléctrica, debido a que actualmente no se cuenta con programas adecuados para realizar dicho control; se obtuvo la información necesaria y se la procedió a organizar adecuadamente.

Así mismo se usó el método inductivo el que admitió la posibilidad de que a partir del análisis de cada uno de los inconvenientes que se ven presentes en las redes de distribución de

Energía Eléctrica, se pueda obtener un problema general de estudio. Y así enfocarnos directamente en resolver dicho problema. Además sirvió a lo largo del proyecto para poder clasificar información que se obtuvo.

Este informe final está formado por secciones bien diferenciadas y varios anexos con información adicional para completar los apartados principales.

En la sección D de Revisión Literaria, se da una visión general de los conceptos, herramientas y lenguaje que se han utilizado para llevar a cabo este proyecto fin de carrera. Está divido en tres capítulos, en el capítulo I se describe el Lenguaje Unificado de Modelamiento (UML), en el capítulo II se habla sobre la Metodología ICONIX, en el capítulo III se describe los Antecedentes de la EERSSA y las Redes Eléctricas.

En la sección E, se describe la Metodología, Métodos y las Técnicas, se detallan las actividades realizadas en cada etapa de software.

Seguidamente en la sección F de Resultados, se detallaron aspectos del desarrollo de la aplicación del módulo de “DISEÑO Y CREACIÓN DEL MÓDULO DE PRONÓSTICO DE CARGA, PARA EL DESARROLLO DE UN SOFTWARE PARA EL ESTUDIO DE REDES DE DISTRIBUCIÓN ELÉCTRICA DE LA CIUDAD DE LOJA”.

En la sección G de Discusión, se explica el desarrollo de la propuesta alternativa, describiendo como se alcanzó cada objetivo planteado.

Para finalizar en las secciones siguientes, se expondrá las conclusiones y recomendaciones a base de la experiencia adquirida durante la realización del proyecto de tesis. Se presenta la Bibliografía de los recursos consultados y finalmente constan los Anexos, donde se agrega la información complementaria y necesaria para la ejecución del mismo.

d. REVISIÓN LITERARIA

1. UML

Lenguaje Unificado de Modelado [UML] es un lenguaje para especificar, construir, visualizar y documentar los artefactos de un sistema de software orientado a objetos (OO). Un artefacto es una información que es utilizada o producida mediante un proceso de desarrollo de software.

UML se quiere convertir en un lenguaje estándar con el que sea posible modelar todos los componentes del proceso de desarrollo de aplicaciones. Sin embargo, hay que tener en cuenta un aspecto importante del modelo: no pretende definir un modelo estándar de desarrollo, sino únicamente un lenguaje de modelado. En UML los procesos de desarrollo son diferentes según los distintos dominios de trabajo; no puede ser el mismo el proceso para crear una aplicación en tiempo real, que el proceso de desarrollo de una aplicación orientada a gestión, por poner un ejemplo.

Las diferencias son muy marcadas y afectan a todas las fases del proceso. El método del UML recomienda utilizar los procesos que otras metodologías tienen definidos.

A partir del año 1994, Grady Booch [Booch96] (precursor de Booch '93) y Jim Rumbaugh (creador de OMT) se unen en una empresa común, Rational Software Corporation, y comienzan a unificar sus dos métodos. Un año más tarde, en octubre de 1995, aparece UML (Unified Modeling Language) 0.8, la que se considera como la primera versión del UML. A finales de ese mismo año, Ivan Jacobson, creador de OOSE (Object Oriented Software Engineer) se añade al grupo.

Como objetivos principales de la consecución de un nuevo método que aunara los mejores aspectos de sus predecesores, sus protagonistas se propusieron lo siguiente:

- El método debía ser capaz de modelar no sólo sistemas de software sino otro tipo de sistemas reales de la empresa, siempre utilizando los conceptos de la orientación a objetos (OO).
- Crear un lenguaje para modelado utilizable a la vez por máquinas y por personas.
- Establecer un acoplamiento explícito de los conceptos y los artefactos ejecutables.
- Manejar los problemas típicos de los sistemas complejos de misión crítica.

Lo que se intenta es lograr con esto que los lenguajes que se aplican siguiendo los métodos más utilizados sigan evolucionando en conjunto y no por separado. Y además, unificar las perspectivas entre diferentes tipos de sistemas (no sólo software, sino también en el ámbito de los negocios), al aclarar las fases de desarrollo, los requerimientos de análisis, el diseño, la implementación y los conceptos internos de la OO.

1.1. Modelado De Objetos

En la especificación del UML[1] podemos comprobar que una de las partes que lo componen es un meta modelo formal. Un meta modelo es un modelo que define el lenguaje para expresar otros modelos. Un modelo en OO es una abstracción cerrada semánticamente de un sistema y un sistema es una colección de unidades conectadas que son organizadas para realizar un propósito específico. Un sistema puede ser descripto por uno o más modelos, posiblemente desde distintos puntos de vista.

[1] UML: Lenguaje Unificado de Modelado

Una parte del UML define, entonces, una abstracción con significado de un lenguaje para expresar otros modelos (es decir, otras abstracciones de un sistema, o conjunto de unidades conectadas que se organizan para conseguir un propósito). Lo que en principio puede parecer complicado no lo es tanto si pensamos que uno de los objetivos del UML es llegar a convertirse en una manera de definir modelos, no sólo establecer una forma de modelo, de esta forma simplemente estamos diciendo que UML, además, define un lenguaje con el que podemos abstraer cualquier tipo de modelo.

El UML es una técnica de modelado de objetos y como tal supone una abstracción de un sistema para llegar a construirlo en términos concretos.

El modelado no es más que la construcción de un modelo a partir de una especificación.

Un modelo es una abstracción de algo, que se elabora para comprender ese algo antes de construirlo. El modelo omite detalles que no resultan esenciales para la comprensión del original y por lo tanto facilita dicha comprensión [1].

Los modelos se utilizan en muchas actividades de la vida humana: antes de construir una casa el arquitecto utiliza un plano, los músicos representan la música en forma de notas musicales, los artistas pintan sobre el lienzo con carboncillos antes de empezar a utilizar los óleos, etc. Unos y otros abstraen una realidad compleja sobre unos bocetos, modelos al fin y al cabo. La OMT, por ejemplo, intenta abstraer la realidad utilizando tres clases de modelos OO: el modelo de objetos, que describe la estructura estática; el modelo dinámico, con el que describe las relaciones temporales entre objetos; y el modelo funcional que describe las relaciones funcionales entre valores. Mediante estas tres fases de construcción de modelos, se consigue una abstracción de la realidad que tiene en sí misma información sobre las principales características de ésta.

Los modelos además, al no ser una representación que incluya todos los detalles de los originales, permiten probar más fácilmente los sistemas que modelan y determinar los errores. Según se indica en la Metodología OMT (Rumbaugh), los modelos permiten una mejor comunicación con el cliente por distintas razones:

Es posible enseñar al cliente una posible aproximación de lo que será el producto final. Proporcionan una primera aproximación al problema que permite visualizar cómo quedará el resultado.

Reducen la complejidad del original en subconjuntos que son fácilmente tratables por separado.

Se consigue un modelo completo de la realidad cuando el modelo captura los aspectos importantes del problema y omite el resto. Los lenguajes de programación que estamos acostumbrados a utilizar no son adecuados para realizar modelos completos de sistemas reales porque necesitan una especificación total con detalles que no son importantes para el algoritmo que están implementando. En OMT se modela un sistema desde tres puntos de vista diferentes donde cada uno representa una parte del sistema y una unión lo describe de forma completa. En esta técnica de modelado se utilizó una aproximación al proceso de implementación de software habitual donde se utilizan estructuras de datos (modelo de objetos), las operaciones que se realizan con ellos tienen una secuencia en el tiempo (modelo dinámico) y se realiza una transformación sobre sus valores (modelo funcional).

UML utiliza parte de este planteamiento obteniendo distintos puntos de vista de la realidad que modela mediante los distintos tipos de diagramas que posee. Con la creación del UML se persigue obtener un lenguaje que sea capaz de abstraer cualquier tipo de sistema, sea informático o no, mediante los diagramas, es decir, mediante representaciones gráficas que

contienen toda la información relevante del sistema. Un diagrama es una representación gráfica de una colección de elementos del modelo, que habitualmente toma forma de grafo donde los arcos que conectan sus vértices son las relaciones entre los objetos y los vértices se corresponden con los elementos del modelo. Los distintos puntos de vista de un sistema real que se quieren representar para obtener el modelo se dibuja dé forma que se resaltan los detalles necesarios para entender el sistema.

1.2. Artefactos Para El Desarrollo De Proyectos

Un artefacto es una información que es utilizada o producida mediante un proceso de desarrollo de software. Pueden ser artefactos un modelo, una descripción o un software. Los artefactos de UML se especifican en forma de diagramas, éstos, junto con la documentación sobre el sistema constituyen los artefactos principales que el modelador puede observar.

Se necesita más de un punto de vista para llegar a representar un sistema. UML utiliza los diagramas gráficos para obtener estos distintos puntos de vista de un sistema:

- Diagramas de Implementación.
- Diagramas de Comportamiento o Interacción.
- Diagramas de Casos de uso.
- Diagramas de Clases.

Ejemplo de algunos de los diagramas que utiliza UML.

1.2.1. Diagramas de Implementación

Se derivan de los diagramas de proceso y módulos de la metodología de Booch, aunque presentan algunas modificaciones. Los diagramas de implementación muestran los aspectos físicos del sistema. Incluyen la estructura del código fuente y la implementación, en tiempo de implementación.

Existen dos tipos:

- Diagramas de componentes
- Diagrama de plataformas despliegue

1.2.2. Diagramas De Componentes

Muestra la dependencia entre los distintos componentes de software, incluyendo componentes de código fuente, binario y ejecutable. Un componente es un fragmento de código software (un fuente, binario o ejecutable) que se utiliza para mostrar dependencias en tiempo de compilación.

1.2.3. Diagrama De Plataformas O Despliegue

Muestra la configuración de los componentes hardware, los procesos, los elementos de procesamiento en tiempo de ejecución y los objetos que existen en tiempo de ejecución. En este tipo de diagramas intervienen nodos, asociaciones de comunicación, componentes dentro de los nodos y objetos que se encuentran a su vez dentro de los componentes. Un nodo es un objeto físico en tiempo de ejecución, es decir una máquina que se compone habitualmente de, por lo menos, memoria y capacidad de procesamiento, a su vez puede estar formada por otros componentes [2].

1.2.4. Diagramas De Interacción O Comportamiento

Muestran las interacciones entre objetos ocurridas en un escenario (parte) del sistema. Hay varios tipos:

- Diagrama de secuencia.
- Diagrama de colaboración.
- Diagrama de estado.
- Diagrama de actividad.

1.2.5. Diagrama de secuencia

Muestran las interacciones entre un conjunto de objetos, ordenadas según el tiempo en que tienen lugar. En los diagramas de este tipo intervienen objetos, que tienen un significado parecido al de los objetos representados en los diagramas de colaboración, es decir son instancias concretas de una clase que participa en la interacción. El objeto puede existir sólo durante la ejecución de la interacción, se puede crear o puede ser destruido durante la ejecución de la interacción. Un diagrama de secuencia representa una forma de indicar el período durante el que un objeto está desarrollando una acción directamente o a través de un procedimiento.

En este tipo de diagramas también intervienen los mensajes, que son la forma en que se comunican los objetos: el objeto origen solicita (llama a) una operación del objeto destino. Existen distintos tipos de mensajes según cómo se producen en el tiempo: simples, síncronos, y asíncronos.

Los diagramas de secuencia permiten indicar cuál es el momento en el que se envía o se completa un mensaje mediante el tiempo de transición, que se especifica en el diagrama.

1.2.6. Diagrama de colaboración

Muestra la interacción entre varios objetos y los enlaces que existen entre ellos. Representa las interacciones entre objetos organizadas alrededor de los objetos y sus vinculaciones. A diferencia de un diagrama de secuencias, un diagrama de colaboraciones muestra las relaciones entre los objetos, no la secuencia en el tiempo en que se producen los mensajes. Los diagramas de secuencias y los diagramas de colaboraciones expresan información similar, pero en una forma diferente.

Formando parte de los diagramas de colaboración nos encontramos con objetos, enlaces y mensajes. Un objeto es una instancia de una clase que participa como una interacción, existen objetos simples y complejos. Un objeto es activo si posee un thread o hilo de control y es capaz de iniciar la actividad de control, mientras que un objeto es pasivo si mantiene datos pero no inicia la actividad.

Un enlace es una instancia de una asociación que conecta dos objetos de un diagrama de colaboración. El enlace puede ser reflexivo si conecta a un elemento consigo mismo. La existencia de un enlace entre dos objetos indica que puede existir un intercambio de mensajes entre los objetos conectados.

Los diagramas de interacción indican el flujo de mensajes entre elementos del modelo, el flujo de mensajes representa el envío de un mensaje desde un objeto a otro si entre ellos existe un enlace. Los mensajes que se envían entre objetos pueden ser de distintos tipos, también según como se producen en el tiempo; existen mensajes simples, sincrónicos, balking, timeout y asíncronos. Durante la ejecución de un diagrama de colaboración se crean y destruyen objetos y enlaces.

1.2.7. Diagramas De Actividad

Son similares a los diagramas de flujo de otras metodologías OO[2]. En realidad se corresponden con un caso especial de los diagramas de estado donde los estados son estados de acción (estados con una acción interna y una o más transiciones que suceden al finalizar esta acción, o lo que es lo mismo, un paso en la ejecución de lo que será un procedimiento) y las transiciones vienen provocadas por la finalización de las acciones que tienen lugar en los estados de origen. Siempre van unidos a una clase o a la implementación de un caso de uso o de un método (que tiene el mismo significado que en cualquier otra metodología OO). Los diagramas de actividad se utilizan para mostrar el flujo de operaciones que se desencadenan en un procedimiento interno del sistema.

1.2.8. Diagramas De Estado

Representan la secuencia de estados por los que un objeto o una interacción entre objetos pasa durante su tiempo de vida en respuesta a estímulos (eventos) recibidos. Representa lo que podemos denominar en conjunto una máquina de estados. Un estado en UML es cuando un objeto o una interacción satisfacen una condición, desarrolla alguna acción o se encuentra esperando un evento.

Cuando un objeto o una interacción pasa de un estado a otro por la ocurrencia de un evento se dice que ha sufrido una transición, existen varios tipos de transiciones entre objetos: simples (normales y reflexivas) y complejas. Además una transición puede ser interna si el estado del que parte el objeto o interacción es el mismo que al que llega, no se provoca un cambio de estado y se representan dentro del estado, no de la transición. Como en todas

[2] OO: Orientado a Objetos

las metodologías (OO) se envían mensajes, en este caso es la acción de la que puede enviar mensajes a uno o varios objetos destino

1.2.9. Diagramas De Casos De Uso

Unos casos de uso es una secuencia de transacciones que son desarrolladas por un sistema en respuesta a un evento que inicia un actor sobre el propio sistema. Los diagramas de casos de uso sirven para especificar la funcionalidad y el comportamiento de un sistema mediante su interacción con los usuarios y/o otros sistemas. O lo que es igual, un diagrama que muestra la relación entre los actores y los casos de uso en un sistema. Una relación es una conexión entre los elementos del modelo, por ejemplo la relación y la generalización son relaciones.

Los diagramas de casos de uso se utilizan para ilustrar los requerimientos del sistema al mostrar cómo reacciona una respuesta a eventos que se producen en el mismo. En este tipo de diagrama intervienen algunos conceptos nuevos: un actor es una entidad externa al sistema que se modela y que puede interactuar con él; un ejemplo de actor podría ser un usuario o cualquier otro sistema. Las relaciones entre casos de uso y actores pueden ser las siguientes:

- Un actor se comunica con un caso de uso.
- Un caso de uso extiende otro caso de uso.
- Un caso de uso usa otro caso de uso

1.2.10. Diagramas De Clases

Los diagramas de clases representan un conjunto de elementos del modelo que son estáticos, como las clases y los tipos, sus contenidos y las relaciones que se establecen entre ellos.

Algunos de los elementos que se pueden clasificar como estáticos son los siguientes:

- ***Paquete:*** Es el mecanismo de que dispone UML para organizar sus elementos en grupos, se representa un grupo de elementos del modelo. Un sistema es un único paquete que contiene el resto del sistema, por lo tanto, un paquete debe poder anidarse, permitiéndose que un paquete contenga otro paquete.
- ***Clases:*** Una clase representa un conjunto de objetos que tienen una estructura, un comportamiento y unas relaciones con propiedades parecidas. Describe un conjunto de objetos que comparte los mismos atributos, operaciones, métodos, relaciones y significado. En UML una clase es una implementación de un tipo. [3]

  Los componentes de una clase son:

- ***Atributo:*** Se corresponde con las propiedades de una clase o un tipo. Se identifica mediante un nombre. Existen atributos simples y complejos.
- ***Operación.*** También conocido como método, es un servicio proporcionado por la clase que puede ser solicitado por otras clases y que produce un comportamiento en ellas cuando se realiza [4].

Las clases pueden tener varios parámetros formales, son las clases denominadas plantillas. Sus atributos y operaciones vendrán definidos según sus parámetros formales. Las plantillas pueden tener especificados los valores reales para los parámetros formales,

entonces reciben el nombre de clase parametrizada instanciada. Se puede usar en cualquier lugar en el que se podría aparecer su plantilla.

Relacionando con las clases nos encontramos con el término utilidad, que se corresponde con una agrupación de variables y procedimientos globales en forma de declaración de clase, también puede definirse como un estereotipo (o nueva clase generada a partir de otra ya existente) de un tipo que agrupa variables globales y procedimientos en una declaración de clase. Los atributos y operaciones que se agrupan en una utilidad se convierten en variables y operaciones globales. Una utilidad no es fundamental para el modelado, pero puede ser conveniente durante la programación.

- ***Metaclase:*** Es una clase cuyas instancias son clases. Sirven como depósito para mantener las variables de clase y proporcionan operaciones (método de clase) para inicializar estas variables. Se utilizan para construir metamodelos (modelos que se utilizan para definir otros modelos).
- ***Tipos:*** Es un descriptor de objetos que tiene un estado abstracto y especificaciones de operaciones pero no su implementación. Un tipo establece una especificación de comportamiento para las clases.
- ***Interfaz:*** Representa el uso de un tipo para describir el comportamiento visible externamente de cualquier elemento del modelo.
- ***Relación entre clases:*** Las clases se relacionan entre sí de distintas formas, que marcan los tipos de relaciones existentes:

Asociación: es una relación que describe un conjunto de vínculos entre clases. Pueden ser binarias o n-arias, según se implican a dos clases o más. Las relaciones de asociación vienen identificadas por los roles, que son los nombres que indican el comportamiento que tienen los tipos o las clases, en el caso del rol de asociación (existen otros tipos de roles

según la relación a la que identifiquen). Indican la información más importante de las asociaciones.

Es posible indicar el número de instancias de una clase que participan en una relación mediante la llamada multiplicidad.

Cuando la multiplicidad de un rol es mayor que uno, el conjunto de elementos que se relacionan puede estar ordenado. Las relaciones de asociación permiten especificar qué objetos van a estar asociados con otro objeto mediante un calificador.

El calificador es un atributo o conjunto de atributos de una asociación que determina los valores que indican cuales son los valores que se asociarán.

Una asociación se dirige desde una clase a otra (o un objeto a otro), el concepto de navegabilidad se refiere al sentido en el que se recorre la asociación.

Existe una forma especial de asociación, la agregación, que especifica una relación entre las clases donde el llamado "agregado" indica él todo y el "componente" es una parte del mismo.

Composición: es un tipo de agregación donde la relación de posesión es tan fuerte como para marcar otro tipo de relación. Las clases en UML tienen un tiempo de vida determinado, en las relaciones de composición, el tiempo de vida de la clase que es parte del todo (o agregado) viene determinado por el tiempo de vida de la clase que representa el todo, por tanto es equivalente a un atributo, aunque no lo es porque es una clase y puede funcionar como tal en otros casos.

Generalización: cuando se establece una relación de este tipo entre dos clases, una es una Superclase y la otra es una Subclase. La subclase comparte la estructura y el

comportamiento de la superclase. Puede haber más de una clase que se comporte como subclase.

Dependencia: una relación de dependencia se establece entre clases (u objetos) cuando un cambio en el elemento independiente del modelo puede requerir un cambio en el elemento dependiente.

1.2.11. Relación De Refinamiento

Es una relación entre dos elementos donde uno de ellos especifica de forma completa al otro que ya ha sido especificado con cierto detalle.

Nuevas características del UML

Además de los conceptos extraídos de métodos anteriores, se han añadido otros nuevos que vienen a suplir carencias antiguas de la metodología de modelado. Estos nuevos conceptos son los siguientes:

- Definición de estereotipos: un estereotipo es una nueva clase de elemento de modelado que debe basarse en ciertas clases ya existentes en el metamodelo y constituye un mecanismo de extensión del modelo.
- Responsabilidades.
- Mecanismos de extensibilidad: estereotipos, valores etiquetados y restricciones.
- Tareas y procesos.
- Distribución y concurrencia (para modelar por ejemplo ActiveX/DCOM y CORBA).
- Patrones/Colaboraciones.
- Diagramas de actividad (para reingeniería de proceso de negocios)

- Clara separación de tipo, clase e instancia.
- Refinamiento (para manejar relaciones entre niveles de abstracción).
- Interfaces y componentes.

1.2.12. El Proceso De Desarrollo

UML no define un proceso concreto que determine las fases de desarrollo de un sistema, las empresas pueden utilizar UML como el lenguaje para definir sus propios procesos y lo único que tendrán en común con otras organizaciones que utilicen UML serán los tipos de diagramas.

UML es un método independiente del proceso. Los procesos de desarrollo deben ser definidos dentro del contexto donde se van a implementar los sistemas [5].

2. Metodología Iconix

ICONIX es una metodología simplificada en comparación con otras metodologías más tradicionales, que unifica un conjunto de métodos de orientación a objetos con el objetivo de abarcar todo el ciclo de vida de un proyecto. Fue elaborado por Doug Rosenberg y Kendall Scott a partir de una síntesis del proceso unificado de los “Three Amigos” Booch, Rumbaugh y Jacobson y que ha dado soporte y conocimiento a la metodología ICONIX desde 1993. Además ICONIX está adaptado a los patrones de diseño y ofrece el soporte de UML, dirigido por casos de uso (como RUP) y es un proceso iterativo e incremental. Es un proceso pequeño y ligero, como XP, pero no descarta las fases del análisis y del diseño como lo hace XP.

ICONIX se divide en dos flujos de trabajo: Dinámico y Estático, los cuales son altamente interactivos (ver figura 1)

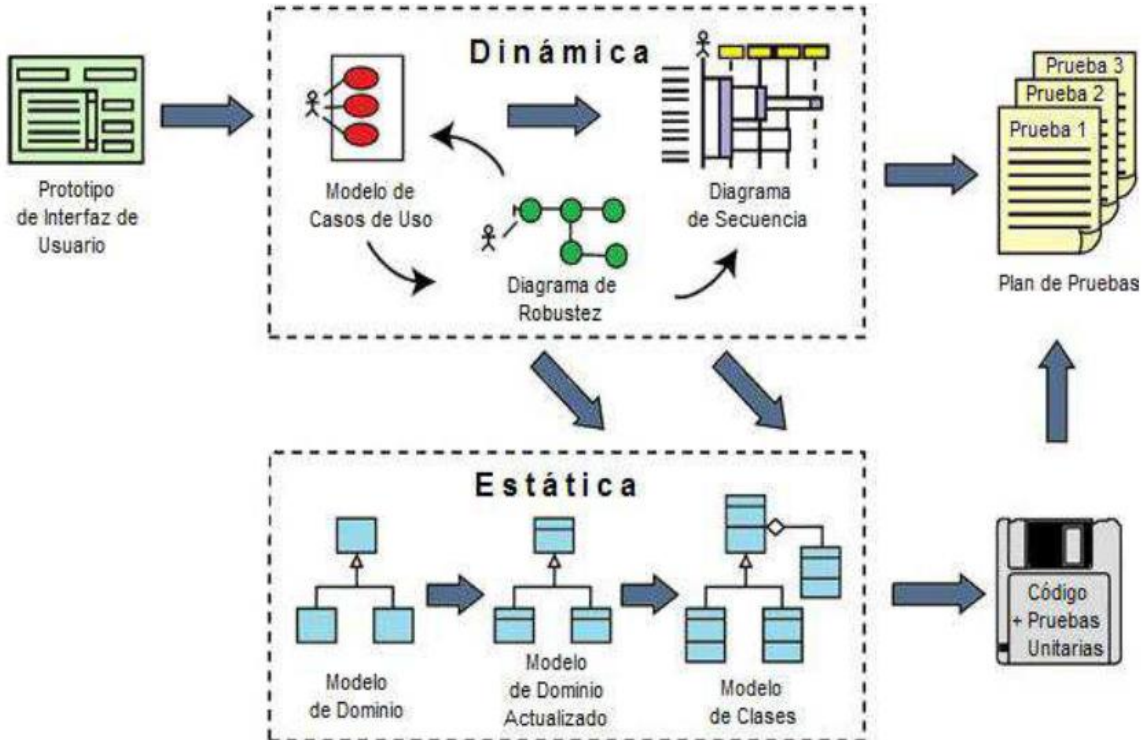


Figura 1. Flujos de trabajo de ICONIX

Las tres características fundamentales de ICONIX son:

- **Iterativo e Incremental:** varias iteraciones ocurres entre el desarrollo del modelo de dominio y la identificación de los casos de uso. El modelo estático es incremental refinado por los modelos dinámicos.

- **Trazabilidad:** cada paso está referenciado por algún requisito. Se define trazabilidad como la capacidad de seguir una relación entre los diferentes artefactos producidos.
- **Dinámica del UML:** la metodología ofrece un uso “dinámico del UML” como los diagramas de casos de uso, diagramas de secuencia, y diagramas de robustez.

Las principales fases de ICONIX son:

- Análisis de Requerimientos
- Análisis y Diseño Preliminar
- Diseño Detallado
- Implementación

2.1. Análisis De Requerimientos

El trabajo es iniciado con un relevamiento informal de todos los requisitos que en principio deberían ser parte del sistema.

2.1.1. Requerimientos Funcionales

Para empezar un proyecto adecuadamente, alguien (posiblemente un equipo de analistas del negocio) hablará con el cliente, usuarios finales, y stakeholders, a fin de obtener un documento con un listado de requerimientos funcionales. La definición de los requerimientos cae ligeramente fuera del alcance de ICONIX, pero se recomienda elaborarla.

Los requisitos funcionales de alto nivel, por lo general (aunque no siempre) empiezan con la frase “El sistema permitirá hacer…, o El sistema no permitirá hacer…,”, y son una fuente importante para crear el modelo de dominio. [6]

2.1.2. Modelado del Dominio

El modelado del dominio es la tarea de construir un glosario del proyecto, o un diccionario de términos (sustantivos, nombres simples o expresiones nominales) utilizados en un proyecto, y mostrar gráficamente como se relacionan entre si todos esos términos, pero en la práctica se trata de una simplificación de diagramas de clases.

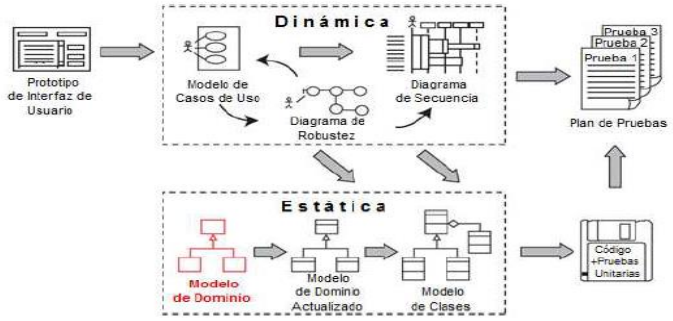


Figura 2. . Modelo de dominio, en contexto del proceso ICONIX

El modelo de dominio define el alcance de un proyecto y forma la base sobre la cual construir los casos de uso. Un modelo, también proporciona un vocabulario común para una comunicación clara entre los miembros de un equipo del proyecto.

El modelo de dominio constituye la parte estática, mientras que los casos de uso son la base de la parte dinámica. La parte estática describe la estructura, y la parte dinámica describe el comportamiento.

Directrices para crear el modelo de dominio

a. Iniciar con el modelo de dominio antes de escribir los casos de uso.
b. En la redacción del listado de requisitos, se debe identificar (subrayar o resaltar) “objetos” que representan cosas y conceptos del mundo real. No confundir un objeto con una tabla de base de datos.
c. Generar un listado ordenado alfabéticamente con todos los términos (nombres de objetos de dominio) identificados en los requerimientos, cambiando todos los términos en plural a singular.
d. Realizar un análisis de todos los términos, a fin de eliminar términos ambiguos o duplicados.
e. Con la lista de objetos de dominio o clase de dominio depurada, realizar un diagrama de clases simplificado, es decir, sin mostrar atributos, ni métodos.

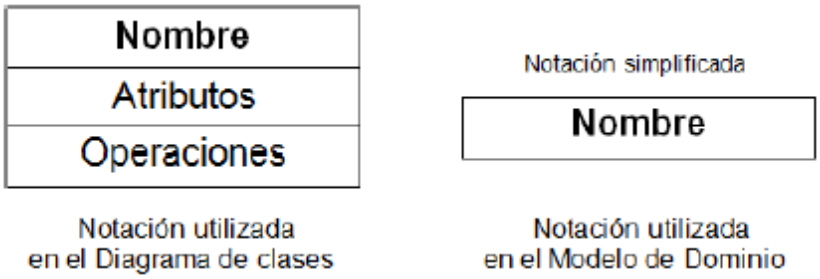


Figura 3. Notación para representar una clase

A medida que se va avance en las fases de ICONIX, se identificara los "atributos" y "operaciones".

f. Usar las relaciones de agregación y generalización para mostrar cómo se asocian los objetos (clases de dominio) entre sí.

ICONIX resalta que casi un 95% de las asociaciones se representan modelándolas como agregación o generalización.

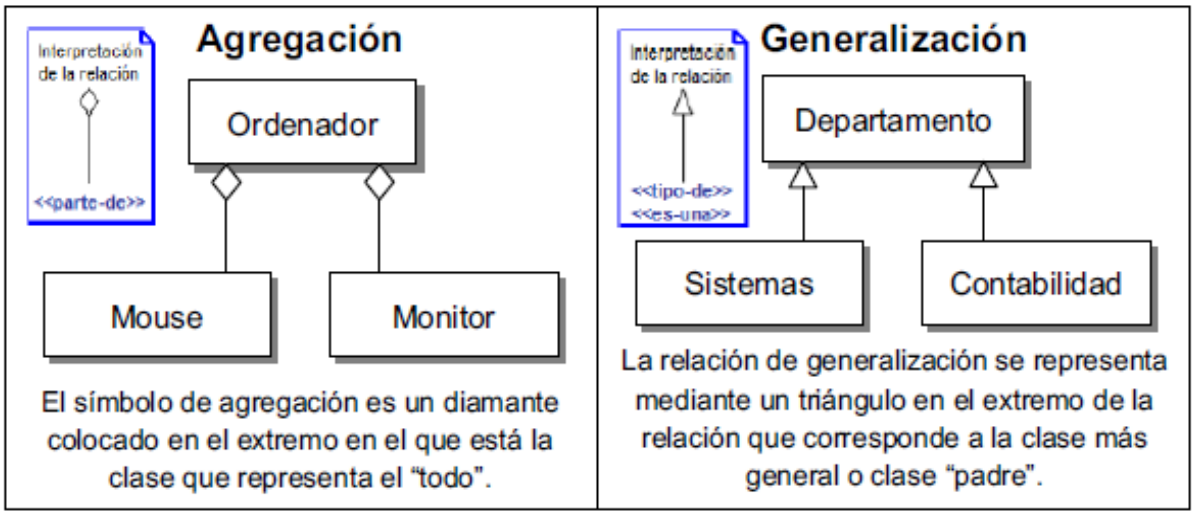


Figura 4. Relaciones para asociar los objetos de dominio

g. Para mostrar cómo se asocian las clases de dominio también se puede utilizar los otros tipos de relaciones (dependencia, asociación, composición).

Un modelo del dominio es una representación de las clases conceptuales del mundo real, no de componentes software.

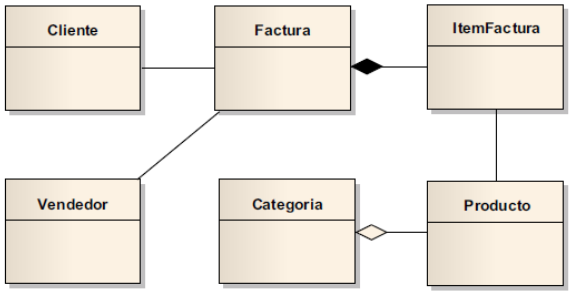


Figura 5. Ejemplo de un modelo de dominio

2.1.3. Modelado de Casos de Uso

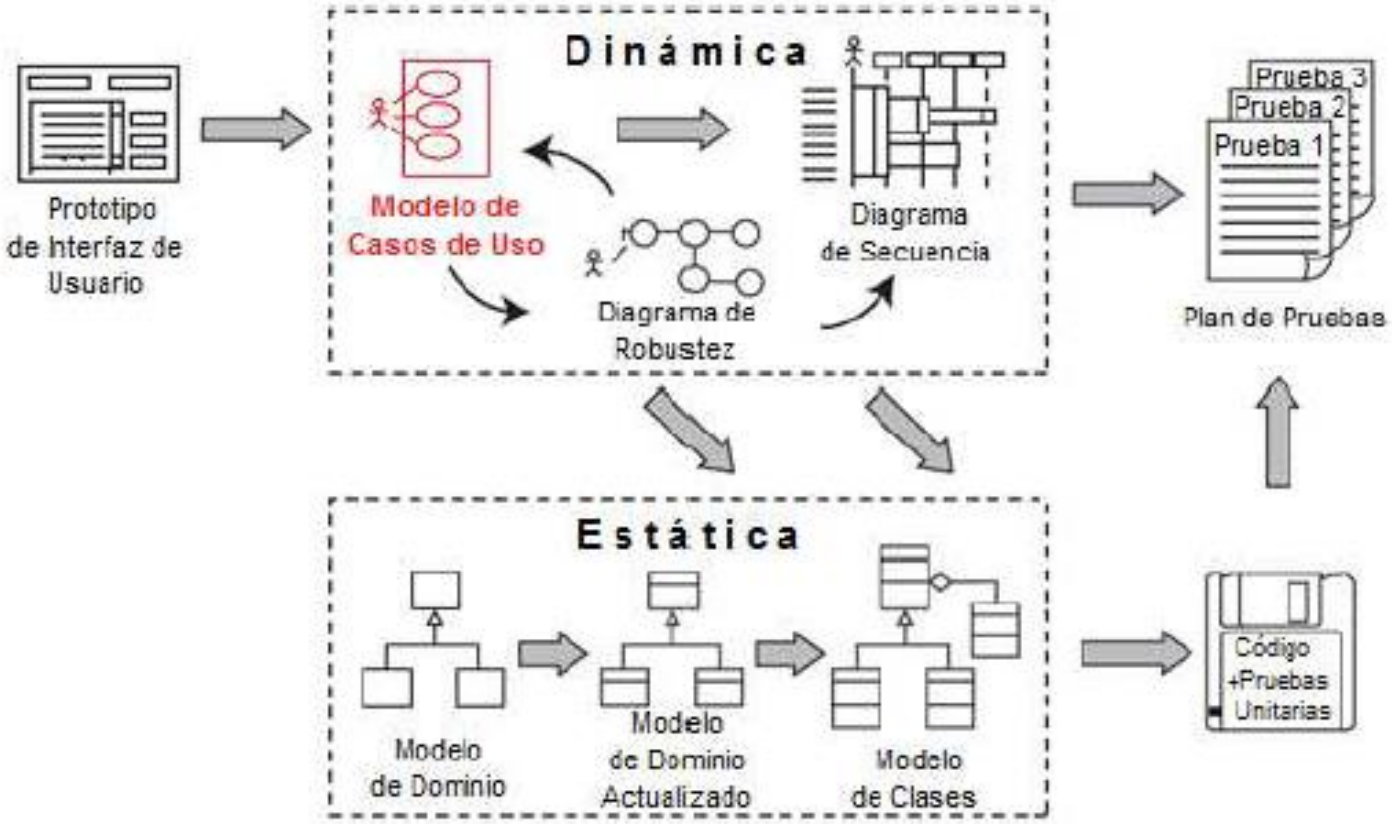


Figura 6. Modelo de casos de uso, en contexto del proceso ICONIX

Con un modelo de dominio inicial se procede a escribir los casos de uso del sistema mostrando los actores involucrados. La propuesta ICONIX supone que el modelo de dominio inicial tiene errores, por tal motivo ICONIX propone una mejora incremental de este modelo a medida que se analice los casos de uso.

También se actualiza el modelo de dominio cuando se dibuja los “diagramas de robustez” y secuencia.

Actores.- un actor es una entidad externa (personas, otros sistemas, sensores, etc.) al sistema que realiza algún tipo de interacción con el mismo. Se representa mediante una figura humana (ver figura 7)

Figura 7. Representación gráfica de un actor

Casos de Uso.- un caso de uso es una descripción de la secuencia de interacciones que se producen entre un actor y el sistema, cuando el actor usa el sistema para llevar a cabo una tarea específica. Se representa mediante una elipse con el nombre del caos de uso en su interior.

Los casos de uso describen bajo la forma de acciones y reacciones el comportamiento de un sistema desde el punto de vista de un usuario. Permiten definir los límites del sistema y las relaciones entre el sistema y el entorno.

Un caso de uso es una manera específica de utilizar un sistema. Es la imagen de una funcionalidad del sistema en respuesta a la estimulación de un actor externo. El conjunto de funcionalidades de un sistema se determina examinando las necesidades funcionales de cada actor. La asociación de un actor con un caso de uso significa responsabilidades.

Directrices para describir los casos de uso

a. El trabajo empieza identificando los casos de uso, a nivel de nombre (verbo o frase verbal corta).

b. Para facilitar la identificación de los casos de uso se recomienda realizar prototipos de interfaz de usuario, maquetas de pantallas, estas son ayudas visuales muy útiles cuando se está escribiendo los casos de uso.

c. Dibujar diagramas de casos de uso: un diagrama de casos de uso muestra un conjunto de casos de uso, actores y sus relaciones. Son importantes en la captura de los requisitos funcionales del sistema.

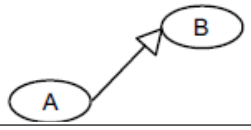
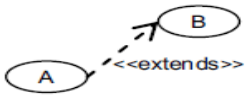
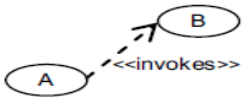
Relación	Representación	Descripción
Generalización		El caso de uso A es un tipo del caso de uso B.
Extensión		Todos los pasos del caso de uso A son efectuados durante la ejecución del caso de uso B, en el punto de extensión especificado en B.
Precede		El caso de uso A debe llevarse a cabo en su totalidad, antes que empiece el caso de uso B. Esto permitirá saber qué caso de uso debe programarse antes que otro.
Invoca		El caso de uso B ocurre durante la vida útil del caso de uso A.

Figura 8. Relaciones para asociar los casos de uso

d. En el modelado de casos de uso no se debe perder mucho tiempo en identificar las relaciones entre los casos de uso, esto se debe porque la parte más importante del trabajo es escribir el texto de los casos de uso.

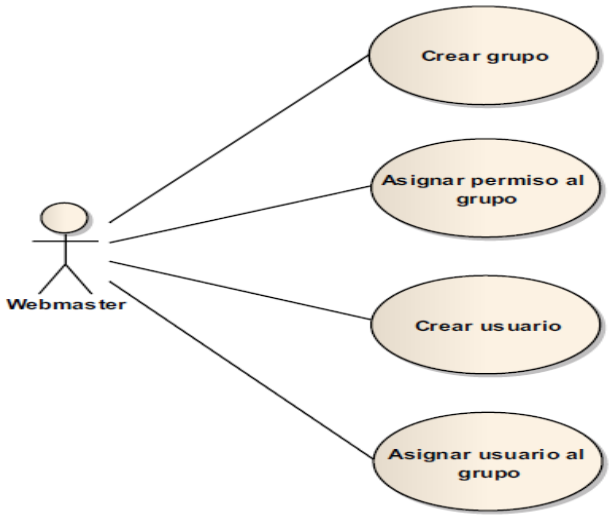


Figura 9. Ejemplo de un diagrama de casos de uso

e. Antes de escribir los casos de uso, se debe organizar todos estos dentro de paquetes, es decir, utilizar los diagramas de paquetes, que son una forma de agrupar elementos fuertemente relacionados (por ejemplo, clases, diagramas, o casos de uso). Una buena práctica para organizar los casos de usos en paquetes, es identificando áreas funcionalmente relacionadas (subsistemas).

f. Escribir los casos de uso, tomando en cuenta las siguientes recomendaciones:

- Escribir los casos de uso en contexto del modelo de dominio, esto es, asociar los requerimientos funcionales con los casos de uso y con los objetos de dominio (trazabilidad).
- Seguir la regla “dos párrafos”, es decir, cada caso de uso debe estar conformado máximo por dos párrafos, incluyendo los cursos básico y alternativo. Si algún caso de uso es demasiado largo resultará incomprensible en el diagrama de secuencia, entonces, si es necesario se debe dividir en dos o más casos de uso separados.

- Escribir los casos de uso en tiempo presente y voz activa, es probable que en la redacción de los requerimientos funcionales se encuentre texto en voz pasiva, entonces, se debe cambiar a voz activa.

- Redactar los casos de uso usando la siguiente estructura: sustantivo-verbo-sustantivo, esto facilitará el diseño de los diagramas de secuencia, ya que estos están orientados alrededor de los sustantivos y verbos:

  - **Los sustantivos.-** Son las instancias de objetos, estos por lo general vienen del modelo de dominio (entidades) o son objetos de la interfaz gráfica de usuario.

  - **Los verbos.-** Son los mensajes entre los objetos, estos representan las funciones del software (controladores) que deben ser construidos.

- Escribir los casos de uso usando flujos de evento/respuesta; cuando se escribe los escenarios de los casos de uso, se está describiendo la interacción del usuario con el sistema.

  En general un caso de uso, constará de varias etapas. Cada paso involucra un evento y una respuesta: la acción del usuario y la reacción del sistema, o viceversa.

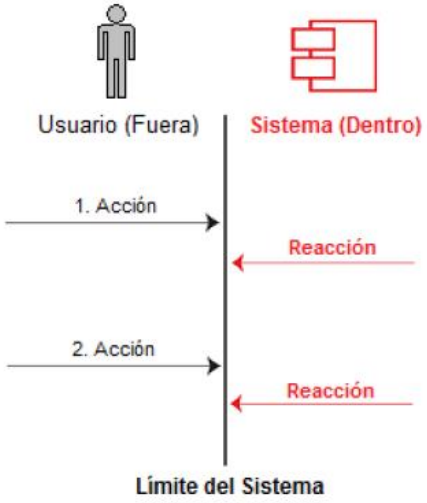


Figura 10. Escenario de un caso de uso

- Si se dispone de manuales de usuario, utilizar estos para extraer los casos de uso.

Para presentar la especificación de cada caso de uso se recomienda utilizar la siguiente plantilla:

Especificación del caso de uso "<<nombre del caso uso>>"

a. **Caso de Uso:** [Clave y nombre del caso de uso.]
b. **Precondiciones:** [Una precondición (de un caso de uso) es el estado del sistema que debe existir antes de que inicie la ejecución del caso de uso, generalmente son aspectos que no van a ser validados durante el caso de uso, sino que se dan por cierto.]

b.1. Precondición 1…
b.2. Precondición 2…

c. **Flujo de Eventos:**

c.1. Flujo Básico: [Debe ser redactado en forma de diálogo entre el actor y el sistema o viceversa.]

c.2. Flujos Alternativos: [Cada flujo alterno representa un comportamiento diferente (muchas veces, debido a las excepciones que pueden ocurrir en el flujo principal).]

c.2.1. Flujo alternativo 1…

c.2.2. Flujo alternativo 2…

d. **Pos condiciones:** [Una pos condición (de un caso de uso) es una lista de posibles estados del sistema que pueden existir después de que termine la ejecución del caso de uso.]

d.1. Pos condición 1…

d.2. Pos condición 2…

Antes de empezar con la siguiente fase Análisis y Diseño Preliminar, se debe garantizar que el modelo de dominio y los casos de uso trabajen juntos, a fin de abordar las necesidades del cliente. Las personas involucradas en este punto son: el cliente, los usuarios finales, gerente de marketing, básicamente todos los interesados en el proyecto.

2.2. Análisis y Diseño Preliminar

Para ir de los casos de uso al diseño detallado (y luego al código), es necesario vincular los casos de uso a los objetos. Una técnica utilizada como puente para ir del análisis al diseño es el Análisis de Robustez.

2.2.1. Análisis de Robustez

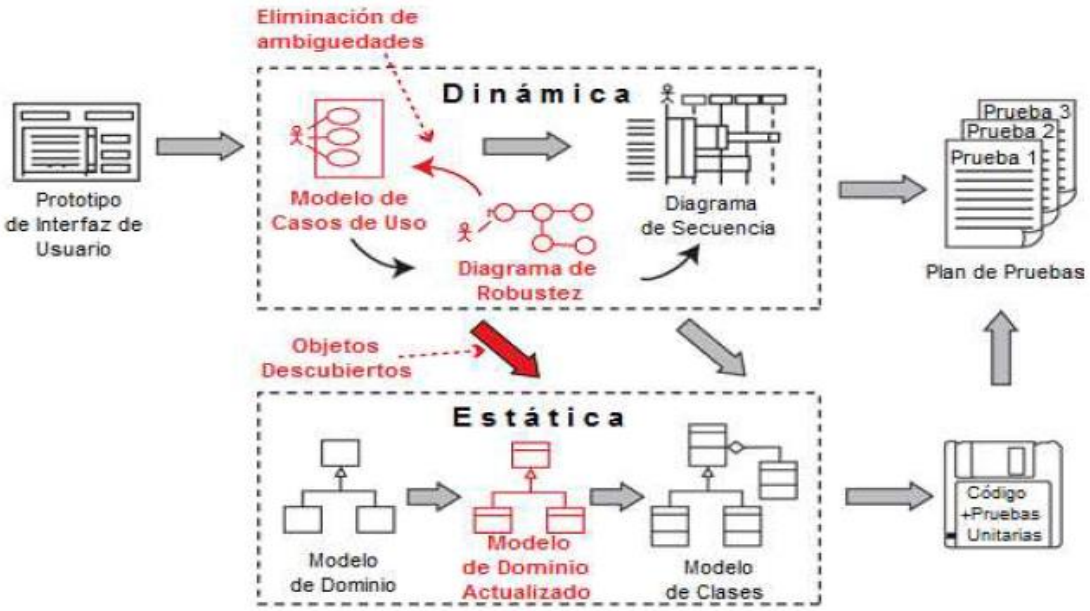


Figura 11. Análisis de Robustez, en contexto del proceso ICONIX

El análisis de robustez facilita el reconocimiento de objetos, sirve para identificar más y mejores clases antes del desarrollo de los diagramas de secuencia. Esta técnica sirve como interfaz entre el “que” y el “como” de un análisis. Además el análisis de robustez provee una gran ayuda para saber si las especificaciones del sistema son razonables [7].

Realizar diagramas de robustez.- Se debe ilustrar gráficamente las interacciones entre los objetos participantes de un caso de uso. Estos diagramas permiten analizar el texto narrativo de cada caso de uso e identificar un conjunto inicial de objetos participantes de cada caso de uso.

Los símbolos utilizados en el diagrama de robustez son:

- **Objetos interfaz (Pantallas).-** Usados por los actores para interactuar con el sistema (es decir, la capa de presentación). Ejemplo: ventanas, diálogos y menús.
- **Objetos entidad (Almacenamientos).-** Son objetos del modelo de dominio.
- **Objetos controladores (Gestores).-** Es la unión entre la interfaz y los objetos entidad. Sirven como conexión entre los usuarios y los datos.

Es útil pensar en los objetos interfaz y los objetos entidad como sustantivos, y los objetos controladores como si fueran verbos.

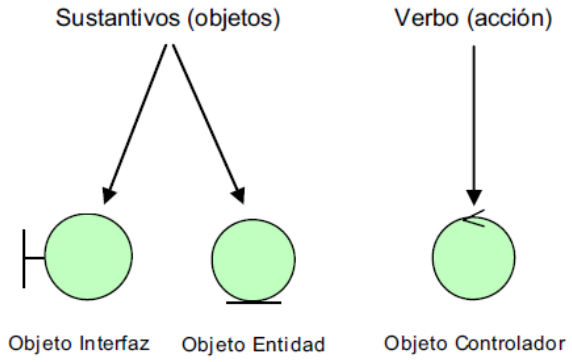


Figura 12. Símbolos utilizados en el Diagrama de Robustez

Reglas básicas para realizar los diagramas de robustez

- Actores solo pueden comunicarse con objetos interfaz [8].
- Las interfaces solo pueden comunicarse con controles y actores.
- Los objetos entidad solo pueden comunicarse con controles.
- Los controles se comunican con interfaces, objetos entidad y con otros controles pero nunca con actores.

Tomando en cuenta que los objetos entidad y la interfaces son sustantivos y los controles son verbos. Se puede enunciar de manera sencilla que los sustantivos nunca se comunican

con otros sustantivos, pero los verbos, si se comunican con otros verbos y a su vez con sustantivos.

Estas reglas ayudan a cumplir el formato sustantivo-verbo-sustantivo sugerido para escribir el texto de los casos de uso.

Directrices para elaborar el análisis de robustez

a. Se debe graficar un diagrama de robustez por cada caso de uso, el diagrama debe contemplar el flujo normal y todos los flujos alternativos.
b. El texto de las etiquetas empleadas en los diagramas de robustez, debe ser tomada del texto de los casos de uso. Debido a que un diagrama de robustez es esencialmente una representación gráfica de un caso de uso.
c. Para nombrar los objetos entidad, emplear el nombre de los objetos de dominio (ver el “Modelo de dominio”).
d. Mientras se dibuja los diagramas de robustez, se debe **“actualizar el modelo de dominio”** de forma incremental, es decir, agregar los nuevas clases de dominio y atributos identificados.
e. Es importante **“rescribir los casos de uso”** en paralelo con la elaboración de los diagramas de robustez. Esto se debe, a que al dibujar un diagrama de robustez, se obliga a trabajar a través del texto del caso de uso, una frase a la vez.
f. Utilizar un objeto interfaz por cada pantalla, esto ayuda a identificar los nombres de pantallas.
g. Tener en cuenta que:
    - Los controladores son funciones lógicas del software.
    - Las interfaces y clases de entidad en un diagrama de robustez, se convertirán en instancias de objetos en un diagrama de secuencia, mientras que los controladores se convertirán en los mensajes.
    - Un diagrama de robustez es un “objeto imagen” de un caso de uso.

- Los “atributos” se pueden descubrir en el texto de los casos de uso, o de los prototipos de interfaz de usuario, o incluso de los requerimientos funcionales.

Antes de empezar con el Diseño Detallado, se debe asegurar que los diagramas de robustez, el Modelo de Dominio y el texto de los casos de uso coincidan entre sí. Se recomienda que no se asigne las operaciones a las clases de dominio, ya que esta es una actividad del Diseño Detallado.

2.2.2. Diseño Detallado

En esta fase se creará un diseño detallado de los casos de uso, por tal motivo, el texto de los casos de uso debe estar completo, correcto y detallado. Además se debería disponer de casi todas las clases de dominio.

El diseño detallado es conocido como la asignación de comportamiento, es decir, la asignación de las funciones del software. Para conducir el diseño detallado se puede utilizar los **diagramas de secuencia**. Tener en cuenta que hay una relación directa entre cada caso de uso, su diagrama de robustez, y los diagramas de secuencia.

2.2.3. Diagrama de secuencia

El diagrama de secuencia es el núcleo del modelo dinámico y muestra todos los cursos alternos que pueden tomar todos los casos de uso, además, muestra interacciones entre objetos según un punto de vista temporal.

A pesar de que a partir de los diagramas de casos de uso y de los diagramas de robustez ya se dispone entre un 75 y 80 por ciento de atributos de las clases identificadas, es hasta los diagramas de secuencia donde se empiezan a ver qué métodos llevaran las clases del sistema.

Notación de un diagrama de secuencia

En el siguiente ejemplo se muestra como está compuesto un diagrama de secuencia.

Directrices para dibujar los diagramas de secuencia

a. Dibujar un diagrama de secuencia por cada caso de uso.

b. Se recomienda pegar el texto (flujo básico y alternativo) del caso de uso en una nota en el margen izquierdo del diagrama de secuencia.

c. En la parte superior del diagrama de secuencia copiar: actores, objetos de tipo interfaz y entidad, identificados en el diagrama de robustez. Los objetos controladores (verbos) del diagrama de robustez se convierten en mensajes.

d. Mostrar cómo el comportamiento del caso de uso es llevada a cabo por los objetos controladores del diagrama de robustez, es decir, mostrar cómo los objetos interactúan mediante el envío de mensajes unos a otros. A menudo un controlador se convierte en dos o más operaciones.

e. Asignar **las operaciones a las clases** mientras se dibuja los mensajes. Es posible detectar nuevos atributos, estos deben incluirse en el diagrama de clases.

2.3. Resumen ICONIX

Con el fin de facilitar la aplicación de la Metodología ICONIX, se presenta un resumen de las fases de ICONIX.

TABLA I RESUMEN DE ICONIX

METODOLOGÍA ICONIX			
Fase	**Actividad**	**Técnica / Notación**	**Entregable/Artefacto**
Análisis de Requerimientos	Definir requerimientos funcionales	Entrevistas, encuestas, con las personas interesadas en el Proyecto.	• Documento: Listado de Requerimientos.
	Modelar el dominio	• Modelo de Dominio. • Lenguaje UML.	• Modelo de Domino Inicial.
	Modelar los casos de uso	• Prototipado. • Diagrama de casos de uso. • Diagrama de paquetes. • Lenguaje UML.	• Prototipo GUI. • Diagramas de Casos de Uso. • Diagrama de Paquetes. • Especificación de casos de uso (Borrador).
Análisis y Diseño Preliminar	Realizar análisis de robustez	• Diagrama de robustez. • Lenguaje UML.	• Diagrama de Robustez por cada caso de uso. • Especificación de casos de uso. • Modelo de dominio actualizado.
Diseño Detallado	Asignar comportamiento	• Diagrama de secuencia. • Lenguaje UML.	• Diagrama de secuencia por cada caso de uso. • Diagrama de Clases. • Modelo de Datos.
Implementación	Generar código	Utilizar seam-gen 2.2.0.GA.	• Código fuente.
	Realizar pruebas funcionales	Interacción con usuario final.	• Software funcionando correctamente.

3. Antecedentes

En la actualidad y desde algún tiempo las falencias en las redes de distribución, se reflejan tanto en la duración como en la frecuencia de las interrupciones del servicio eléctrico, actividad que perjudica y por consiguiente causa molestia y daño a los usuarios. Un sistema de distribución confiable se fundamenta en lograr el menor número de abonados sin servicio eléctrico en caso de presentarse algún tipo de falla; además de un servicio técnico adecuado y oportuno que cumpla con las leyes y reglamentos establecidos, para ello las empresas distribuidoras cuentan con potentes algoritmos destinadas a evaluar y facilitar las maniobras de operación y mantenimiento de las redes de distribución.

Los sistemas eléctricos han cambiado sus estructuras como un intento de lograr sistemas de energía eléctrica más competitivos. La globalización de la industria y las nuevas tecnologías han aumentado la preocupación acerca de las deficiencias en la calidad del servicio eléctrico alrededor del mundo.

El término “Calidad del servicio potencia eléctrica” abarca la calidad del producto (electricidad) y la calidad del servicio. Para que el usuario pueda usar éste, es necesario que la energía eléctrica haya tenido un recorrido previo por cada una de las etapas de Generación, Transmisión, Subtransmisión y Distribución.

En cada país existen regulaciones que tienen la finalidad de asegurar un nivel satisfactorio en la prestación del servicio eléctrico. En Ecuador, el control de calidad del servicio eléctrico lo reglamenta el Consejo Nacional de Electricidad de Ecuador (CONELEC)[3] y el Centro

[3] CONELEC: Consejo Nacional de Electricidad

Nacional de Control de Energía (CENACE) por medio de regulaciones que ambos establecen.

El diseño del sistema de distribución primario de la Empresa Eléctrica Regional del Sur SA (E.E.R.S.SA) basado en calidad de servicio, mediante el análisis de la frecuencia y duración de las interrupciones del servicio eléctrico, plantea un sistema de distribución primario capaz de dar mayor confiabilidad a los usuarios.

Para esto se ha tomado como referencia la regulación No. CONELEC 004/01 de calidad del servicio eléctrico de distribución, en la cual se encuentran los estándares mínimos de calidad y procedimientos técnicos de medición y evaluación a los que se deben someter las empresas distribuidoras del servicio eléctrico [9].

3.1. Redes Eléctricas

Los sistemas de transformación y de distribución local se describen a partir de la subestaciones.

3.1.1. Redes Eléctricas aéreas: Se utilizan postes de diferentes materiales.

3.1.2. Redes Eléctricas subterráneas:

Se canalizan por andenes o vías especiales.

Estas redes alimentan sectores residenciales, comerciales, industriales entre otros dependiendo de las características de demanda y de la topografía.

3.2. Estado Del Arte

Con la globalización de la industria se ha logrado tener en el mundo una mayor competitividad, lo que ha obligado a que los sistemas eléctricos cambien sus estructuras como un intento de logras lograr sistemas de distribución de energía eléctrica más eficientes y competitivos.

En la actualidad y desde algún tiempo las falencias en las redes eléctricas de distribución se reflejan tanto en la duración como en la frecuencia de las interrupciones del servicio eléctrico, lo que perjudica y causa molestia a los usuarios. Un sistema de distribución confiable se fundamenta en lograr el menor número de abonados sin servicio eléctrico en caso de presentarse algún tipo de falla; además de un servicio técnico adecuado y oportuno que cumpla con las leyes y reglamentos establecidos [10].

El término "Calidad del servicio potencia eléctrica" abarca la calidad del producto (electricidad) y la calidad del servicio. Para que el usuario pueda usar éste, es necesario que la energía eléctrica haya tenido un recorrido previo por cada una de las etapas de Generación, Transmisión, Subtransmisión y Distribución.

En cada país existen regulaciones que tienen la finalidad de asegurar un nivel satisfactorio en la prestación del servicio eléctrico. En nuestro país, el control de calidad del servicio eléctrico lo reglamenta el Consejo Nacional de Electricidad del Ecuador (CONELEC) por medio de regulaciones establecidas [11].

En la E.E.R.S.S.A [4]el diseño de los sistemas de distribución primarios está basado en la calidad del servicio, mediante el debido análisis de la frecuencia y duración de las interrupciones del servicio de energía eléctrica, y plantea un sistema de distribución primario

[4] EERSSA: Empresa Eléctrica Regional del Sur S.A.

capaz de dar mayor confiabilidad a los usuarios. Para lograr todo esto se ha tomado como referencia la regulación No. CONELEC –004/01 de calidad del servicio eléctrico de distribución, la misma que contiene los estándares mínimos de calidad y procedimientos técnicos de medición y evaluación a los que se deben someter las empresas distribuidoras del servicio de energía eléctrica.

3.3. INDICADORES CARACTERÍSTICOS EN LA DISTRIBUCIÓN

En los estudios del estado actual de los circuitos de distribución se utilizan varios factores conocidos. A continuación se mencionan los más importantes:

3.3.1. Redistribución de Transformadores

Con el paso del tiempo y por múltiples razones, en las redes de distribución urbanas hay bancos de transformadores que están sobrecargados mientras que otros están subcargados. Un estudio de la cargabilidad de dichos transformadores permite evitar que algunos transformadores pierdan vida útil o se “quemen” mientras que otros estén trabajando ineficientemente por hacerlo muy por debajo de su capacidad [12].

Con esta mejora se aprovecha más la capacidad instalada intercambiando los transformadores subcargados con los sobrecargados o sustituyéndolos, lo que aumenta la eficiencia de la distribución y se liberan capacidades mal utilizadas. Los costos de la reubicación están relacionados con los salarios, el transporte, los gastos técnico administrativos y las pequeñas inversiones que surgen por las roturas y los deterioros detectados en los diferentes equipos que obligan a su sustitución.

La redistribución de transformadores como una medida técnico-económica que las empresas suministradoras deben considerar para utilizar correctamente las capacidades

de los transformadores, ubicados en las redes de distribución primarias, el criterio que propone es el de definir la capacidad mínima del transformador para que durante la vida útil del proyecto, las sobrecargas a las que esté sometido el transformado no dañen su vida útil por sobrecalentamiento, el autor señala que los transformadores están sometidos a un régimen de carga variable durante el día, también se describe que la sobrecarga a que puede estar sometido el equipo sin dañar su aislamiento depende de la forma de la curva de demanda del usuario.

La redistribución de transformadores se realiza de conjunto en una zona o región. El objetivo es intercambiar capacidades entre nodos de la red mientras sea económico pero puede ser conveniente, en algunos casos, comprar nuevos equipos aunque esto obligue a retirar equipos instalados y enviarlos a los almacenes hasta tanto sea necesario en otros lugares. La solución que más convenga a la zona estudiada tiene un planteamiento complejo y se resuelve por técnicas de inteligencia artificial; generalmente utilizando algoritmos genéticos.

3.3.2. Resumen de Métodos y Criterios para la Evaluación de la Red

Se destacan tres aspectos en la literatura estudiada: El cambio de nivel de tensión en las subestaciones o regiones, la reconfiguración de redes y la redistribución zonal de transformadores. El primer aspecto requiere de inversiones considerables y generalmente se realiza en una región cuando el nivel de carga ha aumentado de forma que no es posible dar respuesta eficiente con los niveles de tensión actuales. Por la necesidad de uniformar las existencias y el trabajo con las redes, algunos autores plantean que la tendencia es a realizar cambios de niveles de tensión por regiones o al menos a subestaciones completas.

La reconfiguración de redes requiere de pequeñas inversiones; también se realiza para zonas completas que incluyen grupos de alimentadores vecinos. La topología de la red es decisiva en los complejos estudios de reconfiguración.

La redistribución de transformadores, permite contar con capacidades adecuadas a los niveles de carga de las redes y en lo fundamental produce importantes ahorros de capital a la empresa de servicio eléctrico.

Las dos últimas mejoras, por su complejidad técnica, son estudiadas actualmente utilizando inteligencia artificial; sobre todo algoritmos genéticos.

3.4. Redes De Distribución

La Red de Distribución de la Energía Eléctrica o Sistema de Distribución de Energía Eléctrica es la parte del sistema de suministro eléctrico (ver figura 18), cuya función es el suministro de energía desde la subestación de distribución hasta los usuarios finales (medidor del cliente). Se lleva a cabo por los Operadores del Sistema de Distribución (Distribution System Operator o DSO en inglés).

Los elementos que conforman la red o sistema de distribución son los siguientes:

- Subestación de Distribución Domiciliaria: conjunto de elementos (transformadores, interruptores, seccionadores, etc.) cuya función es reducir los niveles de alta tensión de las líneas de transmisión (o subtransmisión) hasta niveles de media tensión para su ramificación en múltiples salidas.
- Circuito Primario.
- Circuito Secundario.

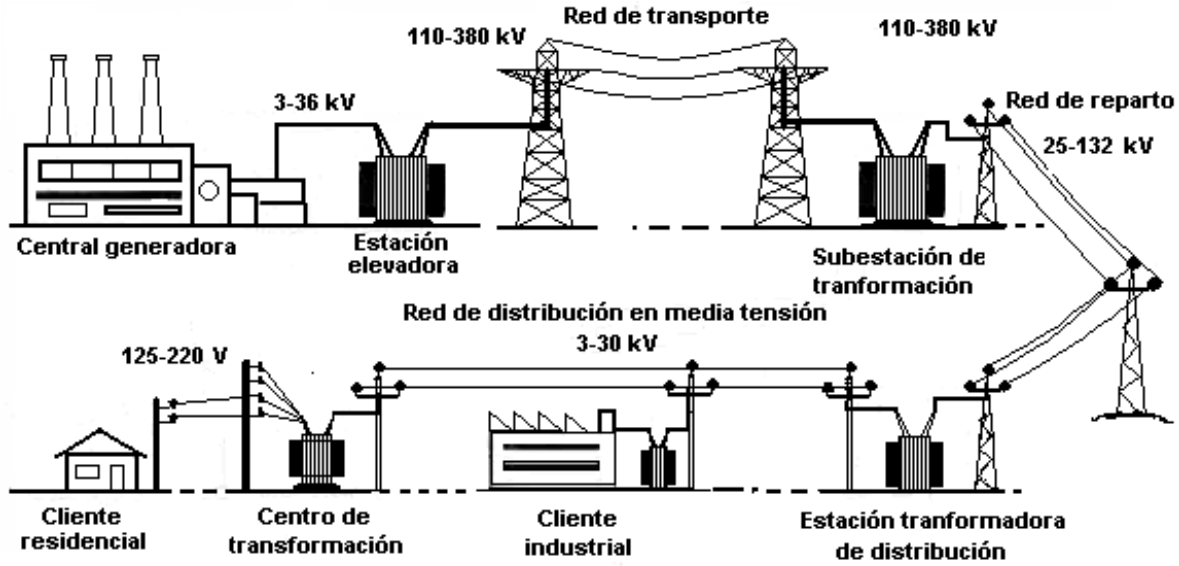


Figura 13. Sistema de Suministro Eléctrico

La distribución de la energía eléctrica desde las subestaciones de transformación de la red de transporte se realiza en dos etapas. La primera está constituida por la red de reparto, que, partiendo de las subestaciones de transformación, reparte la energía, normalmente mediante anillos que rodean los grandes centros de consumo, hasta llegar a las estaciones transformadoras de distribución. Las tensiones utilizadas están comprendidas entre 25 y 132 KV[5]. Intercaladas en estos anillos están las estaciones transformadoras de distribución, encargadas de reducir la tensión desde el nivel de reparto al de distribución en media tensión [13].

La segunda etapa la constituye la red de distribución propiamente dicha, con tensiones de funcionamiento de 3 a 30 KV y con una característica muy radial. Esta red cubre la superficie

[5] KV: Kilovoltio

de los grandes centros de consumo (población, gran industria, etc.), uniendo las estaciones transformadoras de distribución con los centros de transformación, que son la última etapa del suministro en media tensión, ya que las tensiones a la salida de estos centros es de baja tensión (125/220 ó 220/380 V1).

Las líneas que forman la red de distribución se operan de forma radial, sin que formen mallas, al contrario que las redes de transporte y de reparto. Cuando existe una avería, un dispositivo de protección situado al principio de cada red lo detecta y abre el interruptor que alimenta esta red.

La localización de averías se hace por el método de "prueba y error", dividiendo la red que tiene la avería en dos mitades y energizando una de ellas; a medida que se acota la zona con avería, se devuelve el suministro al resto de la red. Esto ocasiona que en el transcurso de localización se pueden producir varias interrupciones a un mismo usuario de la red.

3.4.1. Tipos de sistemas de distribución

Los tipos más conocidos y utilizados para los sistemas de distribución son:

- **Topología de los Sistemas:** esta se encuentra dividida en:

Radial: Estos circuitos reciben el suministro de energía por un solo punto, esa corriente tiene una sola trayectoria desde la subestación hasta los primarios de los transformadores de distribución. La ventaja de este tipo de sistema es su bajo costo en su instalación, una de sus desventajas es que son poco confiables ya que una falla de carácter permanente puede dejar sin servicio a un gran número de consumidores.

Mallada: Posee dos o más caminos simultáneos para el flujo.

Es un sistema más confiable, de mejores características de regulación de tensión; pero es más costoso.

- **Constitución Física:** está dividida en:

Aéreas: En esta modalidad, el conductor que usualmente está desnudo, va soportado a través de aisladores instalados en crucetas, en postes de madera o de concreto.
Las partes principales de un sistema aéreo son esencialmente: postes, conductores, crucetas, aisladores, herrajes, equipos de seccionamiento, transformadores y protecciones en este tipo de redes se emplean transformadores monofásicos con los siguientes valores de potencia o nominales: 10-15-25 - 37.5 - 50 - 75 KVA[6] y para transformadores trifásicos de 30 - 45 - 75 -112.5 y 150 KVA protegidos por cortacircuitos, fusible y pararrayos tipo válvula de 12 KV.

Subterráneas: Son empleadas en zonas donde por razones de urbanismo, estética, congestión o condiciones de seguridad no es aconsejable el sistema aéreo. Actualmente el sistema subterráneo es competitivo frente al sistema aéreo en zonas urbanas céntricas. Los conductores utilizados son aislados de acuerdo a la tensión de operación y conformados por varias capas aislantes y cubiertas protectoras. Estos cables están directamente enterrados o instalados en bancos de ductos (dentro de las excavaciones), con cajas de inspección en intervalos regulares.

Un sistema subterráneo cuenta con los siguientes componentes: ductos, cables, cámaras, empalmes uniones y terminales.

[6] KVA: Kilovoltio Amperio

3.5. Clasificación de los Sistemas Eléctricos de Distribución

Dependiendo de las características de las cargas, los volúmenes de energía involucrados, y las condiciones de confiabilidad y seguridad con que deban operar, los sistemas de distribución se clasifican en:

- Industriales.
- Comerciales.
- Urbana.
- Rural [14].

3.5.1. Sistemas de Distribución Industrial

Comprende a los grandes consumidores de energía eléctrica, tales como las industrias del acero, químicas, petróleo, papel, etc.; que generalmente reciben el suministro eléctrico en alta tensión. Es frecuente que la industria genere parte de su demanda de energía eléctrica mediante procesos a vapor, gas o diésel. Los sistemas de distribución industrial pueden ser tanto de la industria ligera como industria pesada.

En los sistemas de distribución industrial existe gran cantidad de motores instalados, razón por la cual se hace necesario corregir el factor de potencia, además es indispensable distinguir otras cargas como son la calefacción y el alumbrado, a este tipo de cargas se les controla el consumo de reactivos y se les realiza gestión de carga pues tienen doble tarifa para evitar que su pico máximo coincida con el de la carga residencial.

3.5.2. Sistemas de Distribución Comercial

Caracterizados por ser resistivos y se localizan en áreas céntricas de las ciudades donde se realizan actividades comerciales, centros comerciales y edificios de oficinas. Tienen algún componente inductivo que bajan un poco el factor de potencia. Hoy en día predominan cargas muy sensibles que introducen armónicos.

Este tipo de sistemas tienen algunas características especiales como consecuencia de las exigencias en cuanto a seguridad de las personas y de los bienes, por lo que generalmente requieren de importantes fuentes de respaldo en casos de emergencia.

3.5.3. Sistemas de Distribución Urbana

Alimenta la distribución de energía eléctrica a poblaciones y centros urbanos de gran consumo, pero con una densidad de cargas pequeña. Son sistemas en los cuales es muy importante la adecuada selección en los equipos y el dimensionamiento.

Los programas de distribución urbana son desarrollados individualmente por cada empresa de energía y la mayoría de las veces son planes de remodelación y recuperación de pérdidas. Las principales características de las redes de distribución urbana son las siguientes: usuarios muy concentrados; cargas bifilares, trifilares y trifásicas; facilidad de acceso; transformadores generalmente trifásicos en áreas de alta densidad de carga y monofásicos trifilares en áreas de carga moderada.

3.5.4. Sistemas de Distribución Rural

Estos sistemas de distribución se encargan del suministro eléctrico a zonas de menor densidad de cargas, por lo cual requiere de soluciones especiales en cuanto a equipos y a

tipos de red. Debido a las distancias largas y las cargas pequeñas, es elevado el costo del kilo Watt hora (kWh) consumido. En muchos casos es justificado, desde el punto de vista económico, la generación local, en una fase inicial, y sólo en una fase posterior, puede resultar económica y práctica la interconexión para formar una red grande.

3.5.5. Subestación de Distribución

Una subestación es un nodo dentro del sistema de potencia en el cual se transforman los niveles de tensión y corriente con el fin de minimizar pérdidas y optimizar la distribución de la energía producida en las centrales generadoras, maniobrando y controlando su destino final con determinados requisitos de calidad. Los parámetros que la definen son: Potencia, Tensión y Corriente.

La función de una subestación de distribución es similar a la de subtransmisión, en una proporción mucho menor; en este sistema están los circuitos de distribución primarios y los circuitos de distribución secundaria. La distribución primaria recibe la energía de la subestación de distribución y en su recorrido la traspasa directamente a los consumidores medianos, talleres, centros comerciales, etc., y a los consumidores pequeños residenciales a través de la distribución secundaria.

Es frecuente en la distribución primaria tensiones de 4.16 – 7.8 – 13.8 KV. 12.7 – 22 – 34.5 kV, correspondiente en cada caso a las tensiones de fase y de línea de los secundarios de la subestaciones conectadas en la estrella con el neutro sólidamente aterrado, respectivamente este neutro corre simultáneamente con las tres fases, formando un sistema de 4 hilos. Los circuitos de 34.5 y 22 kV pueden cubrir zonas más extensas que los de 13.8 kV por lo que pueden absorber con más facilidad el crecimiento de la carga y la ampliación de las líneas para cubrir las nuevas necesidades de servicio. La distribución está

formada por circuitos trifásicos en sus secciones principales, pero en los ramales de poca carga, generalmente existen 1 ó 2 fases, además del conductor neutro. [15]

3.5.6. Principales Componentes de los Sistemas de Distribución

Los principales elementos componentes de un sistema de distribución son:

3.5.6.1. Alimentadores Primarios de Distribución.

Se considera Alimentadores Primarios a las líneas (conductores) de media tensión que transportan la potencia y energía desde la subestación de distribución hasta cada uno de los transformadores de distribución y acometidas en alta tensión, los conductores van soportados en postes cuando se trata de instalaciones aéreas y en ductos cuando se trata de instalaciones subterráneas. Los componentes de un alimentador primario son:

- **Troncal:** es el tramo de mayor capacidad del alimentador que transmite la energía eléctrica desde la subestación de potencia a los ramales, está formada por circuitos trifásicos.

- **Ramal:** es la parte del alimentador primario energizado a través de un troncal, en el cual van conectados los transformadores de distribución y servicios particulares suministrados en media tensión. Normalmente puede estar conformado de una o dos fases acompañados del neutro de calibre menor a la troncal. Se pueden introducir algunas mejoras en un alimentador tales como:

  - Seccionamiento y / o protección en ramales.
  - Seccionamiento bajo carga.
  - Enlaces.

- Protecciones intermedias.

Los alimentadores primarios normalmente se estructuran en forma radial, en un sistema de este tipo la forma geométrica del alimentador semeja la de un árbol, donde por el grueso del tronco, se transmite el mayor flujo de la energía eléctrica, derivándose la carga a lo largo de los ramales [16].

3.5.6.2. Transformadores de Distribución

En los sistemas de distribución de energía eléctrica, para poder llegar con la energía producida, hasta los consumidores finales, se necesita de un dispositivo llamado transformador. La energía producida en forma masiva, es transmitida desde, la central de generación, hasta una subestación. De éstas, salen los alimentadores primarios hacia las áreas de consumo, en donde se alimentan a los transformadores de distribución, que son los encargados de transmitir la energía hasta los consumidores finales a niveles de tensión convenientes.

3.5.6.3. Clasificación y Utilización de los Transformadores

a. **Por la Operación:** Se refiere a la energía o potencia que manejan dentro del sistema eléctrico de potencia.

   Transformadores de Distribución: Son aquellos transformadores; monofásicos, o trifásicos, que tienen capacidades entre 5 a 500 KVA.

Figura 14. Transformador de distribución

b. **Transformadores de potencia:** Aquellos transformadores con capacidades mayores a 500 KVA.

c. **Por el Número de Fases:** De acuerdo a las características del sistema al cual se conectará, tenemos:

   Transformador monofásico: Son transformadores de potencia o distribución, que son conectados a una línea o fase y un neutro o tierra. Estos transformadores tienen un solo devanado de alta tensión y uno de baja tensión.

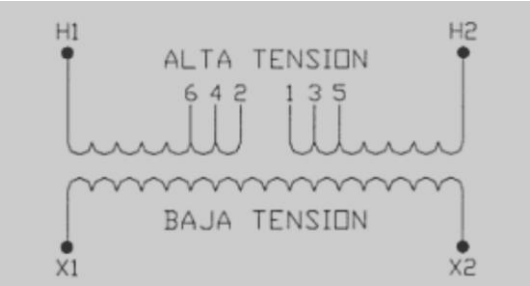


Figura 15. Esquema eléctrico, transformador monofásico

Transformador trifásico: Transformadores de potencia o distribución, que son conectados a tres líneas o fases y pueden estar o no conectados a un neutro común o tierra.

Estos transformadores tienen tres devanados de alta tensión y tres de baja tensión.

Alimentadores Secundarios

Los alimentadores secundarios distribuyen la energía desde los transformadores de distribución hasta las acometidas a los usuarios.

Los conductores de distribución primaria y secundaria corren conjuntamente sobre el mismo poste; los primarios se colocan horizontalmente sobre crucetas en aisladores de porcelana o polímero, mientras que los del secundario se disponen verticalmente directamente sobre el poste en aisladores rollo.

Conductores

Son los elementos metálicos, generalmente cobre o aluminio, permeables al paso de la corriente eléctrica, que cumplen la función de transportar la "presión electrónica" de un extremo a otro del cable. Los metales mencionados se han elegido por su alta conductividad, característica necesaria para optimizar la transmisión de energía [14].

Los conductores pueden estar constituidos por hilos metálicos de distinto diámetro, según la mayor o menor flexibilidad exigida al cable. Los materiales más usado como conductores eléctricos son: el cobre y el aluminio, aunque el primero es superior en características eléctricas y mecánicas (la conductividad del aluminio es aproximadamente un 60% de la del cobre y su resistencia a la tracción es de un 40%), las características de bajo peso y menor costo del aluminio, han dado lugar a un amplio uso tanto para conductores desnudos como aislados.

Figura 16. Conductor de aluminio XLPE aislado y PVC enfundados cable eléctrico

En el cobre usado en conductores eléctricos, se distinguen tres temples: blando, semiduro y duro; con propiedades algo diferentes, siendo el cobre blando de mayor conductibilidad y el cobre duro el de mayor resistencia mecánica [13]

e. MÉTODOS Y TÉCNICAS

En la recolección y organización de la información que se obtenga, para sustentar la presente tesis se hará uso de los siguientes métodos y técnicas:

1. MÉTODOS

1.1. Método Deductivo

La aplicación de este método permitió conocer de manera específica cada uno de los problemas más comunes y que se presentan generalmente en las redes de distribución de energía eléctrica de la EERSSA, en particular el crecimiento de la demanda de energía eléctrica, debido a que actualmente no se cuenta con programas adecuados para realizar dicho control; se obtuvo la información necesaria y se la procedió a organizar adecuadamente para poder plantear la problemática de la presente tesis, además servirá para organizar de la misma manera información posterior que se recolecte.

1.2. Método Inductivo

Este método admitió la posibilidad de que a partir del análisis de cada uno de los inconvenientes que se ven presentes en las redes de distribución de Energía Eléctrica, se pueda obtener un problema general de estudio. Y así

enfocarnos directamente en resolver dicho problema. Además servirá a lo largo del proyecto para poder clasificar información que se pueda obtener.

2. METODOLOGÍA DE DESARROLLO

Para el desarrollo del software se empleará la Metodología ICONIX, la cual permitirá seguir una serie de pasos ordenados y mantener al usuario a la par de los avances que se vayan realizando.

2.1. ESTA METODOLOGÍA ESTÁ COMPUESTA POR CUATRO FASES PRINCIPALES:

2.1.1. Análisis de Requisitos

Se determinarán los objetos del dominio con sus diferentes relaciones, a partir del proceso a automatizar, para lo cual se hará uso de un diagrama de clases, además se identificarán los casos de uso por lo tanto se construirá el modelo de casos de uso, también se establecerá los requerimientos funcionales y no funcionales para cada caso de uso y objetos del dominio; además se realizará un prototipado rápido de las interfaces del sistema.

2.1.2. Análisis y Diseño preliminar

Dentro de esta etapa se realizarán las descripciones de los casos de uso incluyendo los cursos básicos y alternos; seguidamente se construirán los diagramas de robustez, identificación de los objetos que formarán parte del sistema y se realizará una actualización del modelo del dominio que ya se había definido.

2.1.3. Diseño

En esta fase se construirán los diagramas de secuencia identificando los mensajes entre objetos, se adicionará los detalles del diseño en el diagrama de clases y finalmente se verificará que el diseño satisfaga los requisitos encontrados al principio.

2.1.4. Implementación y Pruebas

Se generará el código en Java utilizando el IDE NetBeans 6.8 se desarrollarán los manuales correspondientes y posteriormente se realizarán las pruebas para verificar que la aplicación funcione correctamente.

2.2. Ventajas de Iconix

- ICONIX es un modelo pequeño y firme que no desecha el análisis y el diseño.
- Usa un análisis de robustez que reduce la ambigüedad al describir los casos.
- Es usado en proyectos más ligeros que los usados en RUP, por lo que tiene un mayor campo de aplicabilidad.
- Proporciona suficientes requisitos y documentación de diseño, pero sin parar el análisis.
- Es refinado y actualizado a lo largo del proyecto, por lo que siempre refleja la actual comprensión del problema de espacio.

f. RESULTADOS

1. ANÁLISIS DE REQUERIMIENTOS

ANALISIS DE REQUISITOS

Para desarrollar el modulo “DISEÑO Y CREACIÓN DEL MÓDULO DE PRONÓSTICO DE CARGA, PARA EL DESARROLLO DE UN SOFTWARE PARA EL ESTUDIO DE REDES DE DISTRIBUCIÓN ELÉCTRICA DE LA CIUDAD DE LOJA” se utilizó la metodología ICONIX, la cual nos permite desarrollar fase a fase nuestra aplicación

Reconocimiento de los Actores:

- Administrador

RECOLECCIÓN DE LA INFORMACIÓN

Para realizar la aplicación antes mencionada se elaboró visitas de campo a cada subestación perteneciente a la Empresa Regional del Sur (EERSSA). Además de visitar el departamento técnico de la EERSSA con el objetivo de obtener información sobre el consumo de demanda de energía eléctrica.

REQUERIMIENTOS FUNCIONALES

En esta fase se determinó los requerimientos funcionales y no funcionales del módulo el cual nos sirve para saber que funciones cumple el mismo. Además de recibir el apoyo del encargado del proyecto el Ing. Jorge Carrión mediante charlas y visitas de campo a la EERSSA.

REQUERIMIENTOS FUNCIONALES

TABLA II. REQUERIMIENTOS FUNCIONALES

El Módulo permitirá al Administrador:		
Código	**Descripción**	**Categoría**
RF001	Calcular el método de Regresión, Correlación y Pronóstico.	Oculto
RF002	Escoger el alimentador en tres intervalos Día, mes y año.	Evidente
RF003	Escoger las estaciones con sus alimentadores	Evidente
RF004	Escoger el tiempo de consumo en tres intervalos Día, mes y año.	Evidente
RF005	Calcular el método de regresión ya sea en tres intervalos por Día, Mes y Año.	Evidente
RF006	Calcular el método de correlación y pronóstico en el intervalo de tiempo Año.	Evidente

RF008	Calcular la pendiente, el nivel de incertidumbre, pronóstico de certidumbre en base a la tabla de Student para que el resultado sea más factible.	Oculto
RF008	Escoger P (potencia activa) o Q (potencia reactiva) para definir el cálculo por trimestres.	Evidente
RF009	Encontrar la tendencia como la media por cada etapa o estación del año.	Evidente
RF010	Pronosticar el valor del próximo año en base a los años anteriores ya que el año a pronosticar recién está comenzando.	Evidente
RF011	Calcular la pendiente y la desviación estándar a través del método de regresión.	Evidente
RF012	Calcular la recta logarítmica, el nivel de incertidumbre y la desviación estándar mediante el método de pronóstico.	Evidente
RF013	Calcular la varianza, desviación y el rango de incertidumbre mediante el método de pronóstico.	Evidente
RF014	Calcular el pronóstico óptimo con la exponencial de la recta.	Evidente

REQUERIMIENTOS NO FUNCIONALES

TABLA III. REQUERIMIENTOS NO FUNCIONALES

Código	**Atributo**	**Descripción**
RNF001	Sistema Operativo	El módulo será multiplataforma.
RFN002	Interfaz	Interfaz amigable para el usuario

RFN003	Patrón de Desarrollo	El sistema se desarrollará bajo el patrón Modelo Vista Controlador (MVC).
RFN004	Organizacional	Se debe contar con el registro de consumo de demanda ya sea la potencia activa o reactiva.
RFN005	Lenguaje de Programación	La aplicación será desarrollada en Java con su entorno de progresión NetBeans.

MODELO DEL DOMINIO

En esta fase el modelo del dominio representa conceptos del mundo real el cual nos ayuda a identificar qué relación habrá en nuestro módulo denominado Pronóstico de Carga.

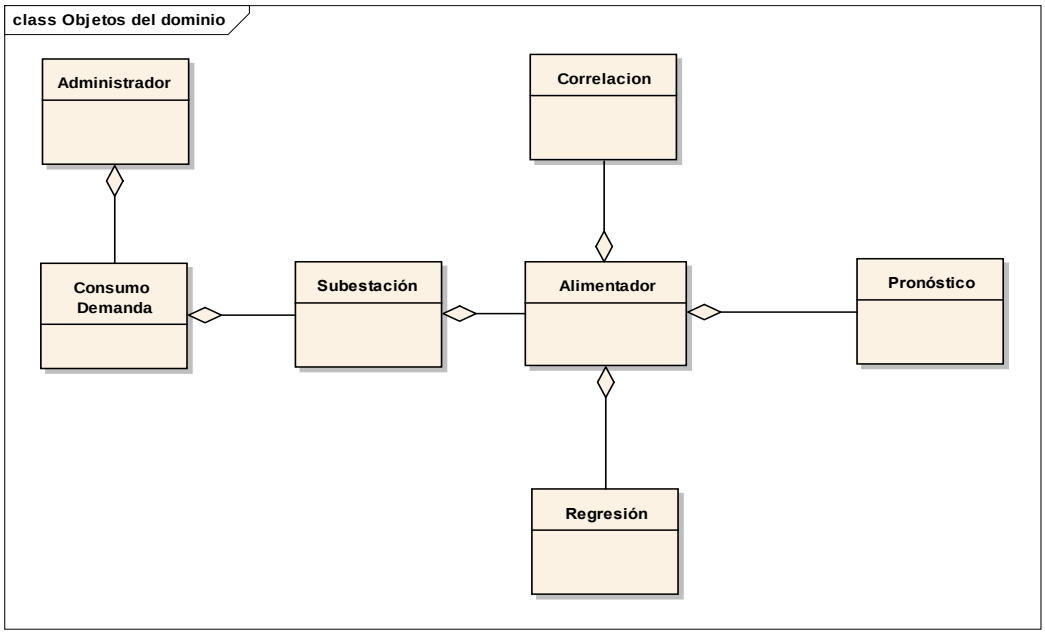


Figura 17. MODELO DEL DOMINIO

TABLA IV. DESCRIPCIÓN MODELO DEL DOMINIO

CLASES	DESCRIPCIÓN
Administrador	Clase que representa al administrador del módulo
Consumo de demanda	Clase que representa el máximo consumo de demanda en tres intervalos Día-Mes-Año.
Subestación	Clase que representa un área de terreno con edificación
Alimentador	Clase que representa un área de terreno con edificación
Regresión	Método para calcular P o Q mediante mínimos cuadrados, ya sea en el intervalo Día-Mes-Año
Correlación	Método para calcular la correlación existente en el intervalo de Mes-Año
Pronóstico	Método que permite pronosticar los valores para el año siguiente con un nivel de certidumbre aceptado usando los trimestres de años anteriores

MODELO DE CASOS DE USO

Los diagramas de casos de uso documentan el comportamiento de un sistema desde el punto de vista del usuario.

Por lo tanto los casos de uso determinan los requisitos funcionales del sistema, es decir, representan las funciones que un sistema puede ejecutar.

El modelo de casos de uso describe la funcionalidad propuesta del nuevo sistema.

Un caso de uso representa una unidad discreta de interacción entre un usuario (humano o máquina) y el sistema.

Un Caso de Uso es una unidad simple de trabajo significativo.

Cada caso de uso tiene una descripción que describe la funcionalidad que se construirá en el sistema propuesto.

Un caso de uso puede "incluir" la funcionalidad de otro caso de uso o "extender" a otro caso de uso con su propio comportamiento.

Módulo de Pronostico de Carga

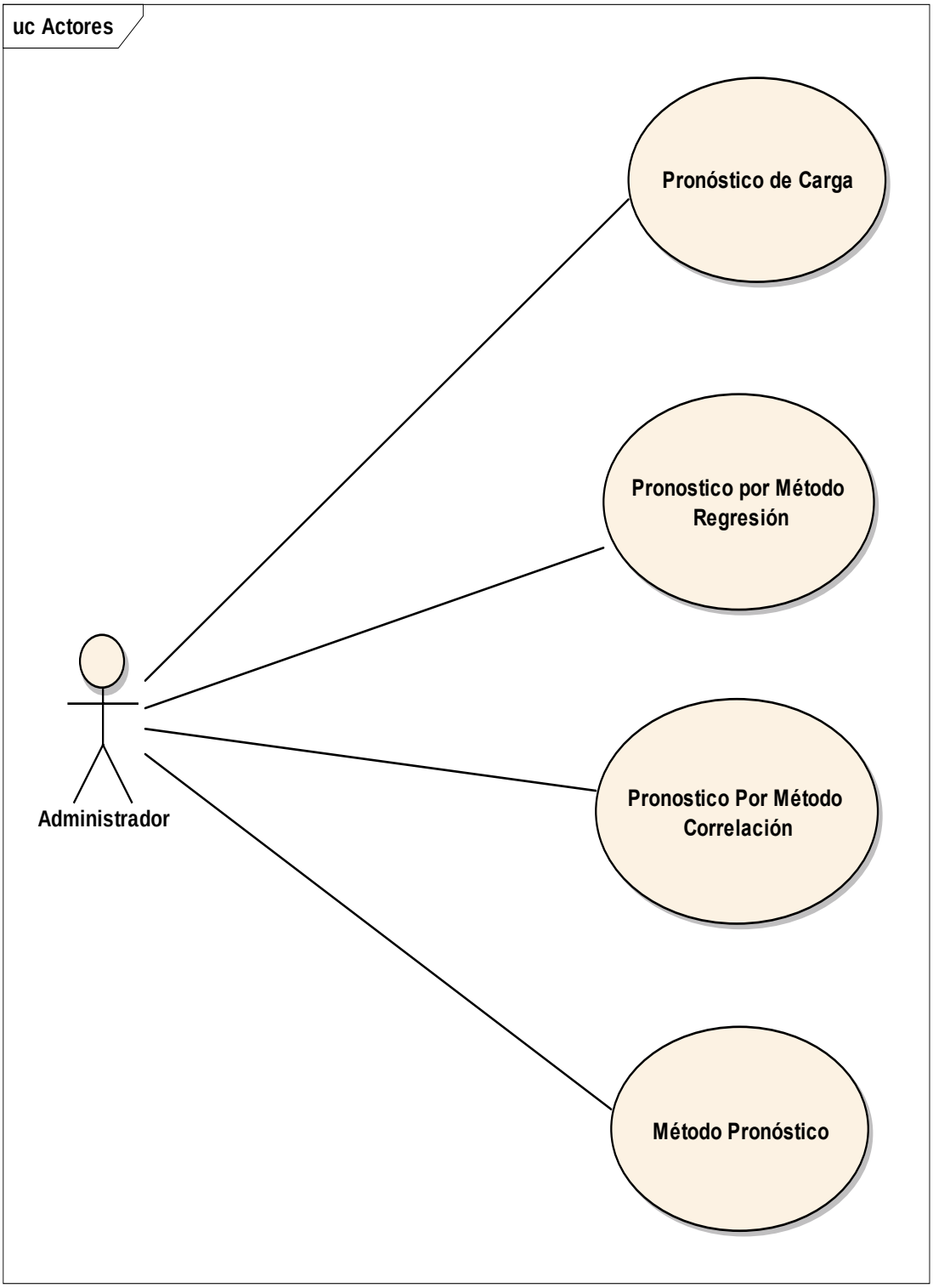


Figura 18. CASO DE USO GENERAL

2. Análisis Y Diseño Preliminar

En esta fase se presenta un prototipo de pantalla de la aplicación y a su vez la descripción del caso de uso del módulo.

2.1. Descripción De Casos De Uso

En esta sección primero hablaremos de los usuarios que intervienen en el módulo de Pronóstico de Carga.

Administrador: es el encargado de controlar los datos que se van a calcular en la aplicación como son el método de regresión, correlación y pronóstico en tres intervalos de tiempo Día, mes y año.

PROTOTIPO INICIAL

En la figura 19: se muestra la pantalla principal la cual nos permite ingresar al menú Reportes y Estadísticas con sus submenús Curvas de Carga, Pronósticos.

Figura 19. PANTALLA PRINCIPAL

En la figura 20, se muestra la pantalla de pronóstico de carga la cual nos permite escoger el intervalo de tiempo ya sea la del día, mes o año.

Además de escoger las Subestaciones y los alimentadores correspondientes a cada subestación.

Figura 20. PRONOSTICO DE CARGA

En la figura 21, se muestra la ventana en donde podemos pronosticar por el método de Regresión.

Esto se lo podrá hacer en un intervalo de día-mes-año dependiendo de las necesidades del administrador

Así mismo escoger el alimentador de la subestación deseada, pudiendo determinar si es P o Q el valor a utilizarse, además de calcular el rango de certidumbre y los grados de libertad.

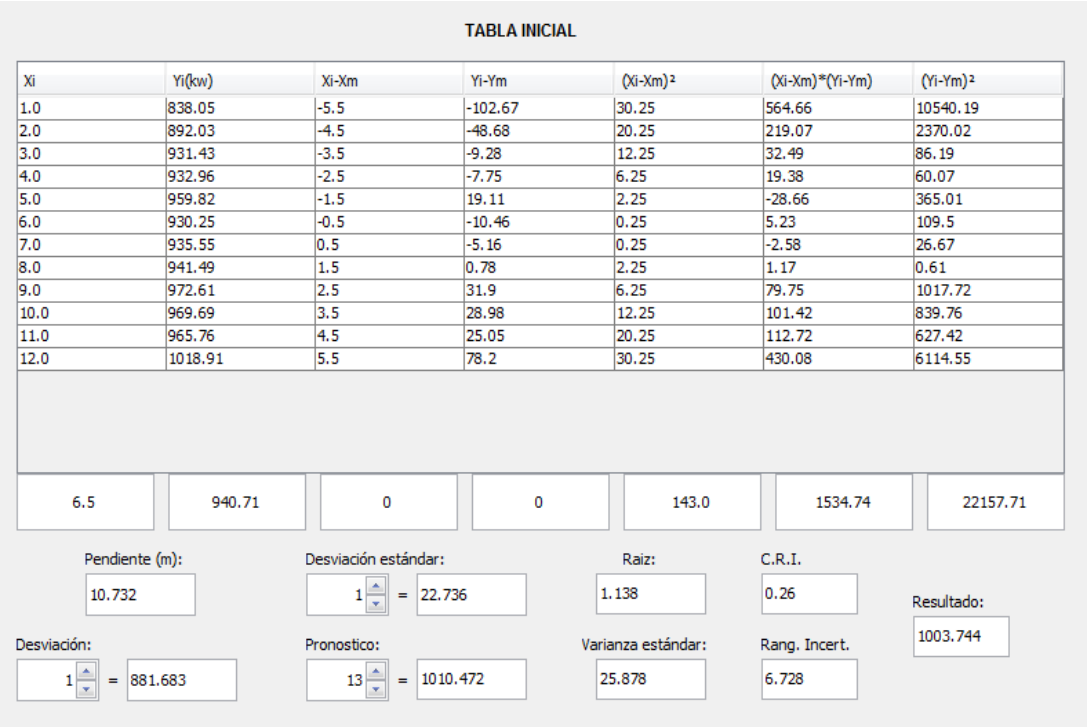


TABLA INICIAL

Xi	Yi(kw)	Xi-Xm	Yi-Ym	(Xi-Xm)²	(Xi-Xm)*(Yi-Ym)	(Yi-Ym)²
1.0	838.05	-5.5	-102.67	30.25	564.66	10540.19
2.0	892.03	-4.5	-48.68	20.25	219.07	2370.02
3.0	931.43	-3.5	-9.28	12.25	32.49	86.19
4.0	932.96	-2.5	-7.75	6.25	19.38	60.07
5.0	959.82	-1.5	19.11	2.25	-28.66	365.01
6.0	930.25	-0.5	-10.46	0.25	5.23	109.5
7.0	935.55	0.5	-5.16	0.25	-2.58	26.67
8.0	941.49	1.5	0.78	2.25	1.17	0.61
9.0	972.61	2.5	31.9	6.25	79.75	1017.72
10.0	969.69	3.5	28.98	12.25	101.42	839.76
11.0	965.76	4.5	25.05	20.25	112.72	627.42
12.0	1018.91	5.5	78.2	30.25	430.08	6114.55
6.5	940.71	0	0	143.0	1534.74	22157.71

Figura 21. MÉTODO DE REGRESIÓN

En la figura 22, se muestra la ventana en donde podemos pronosticar por el método de correlación.

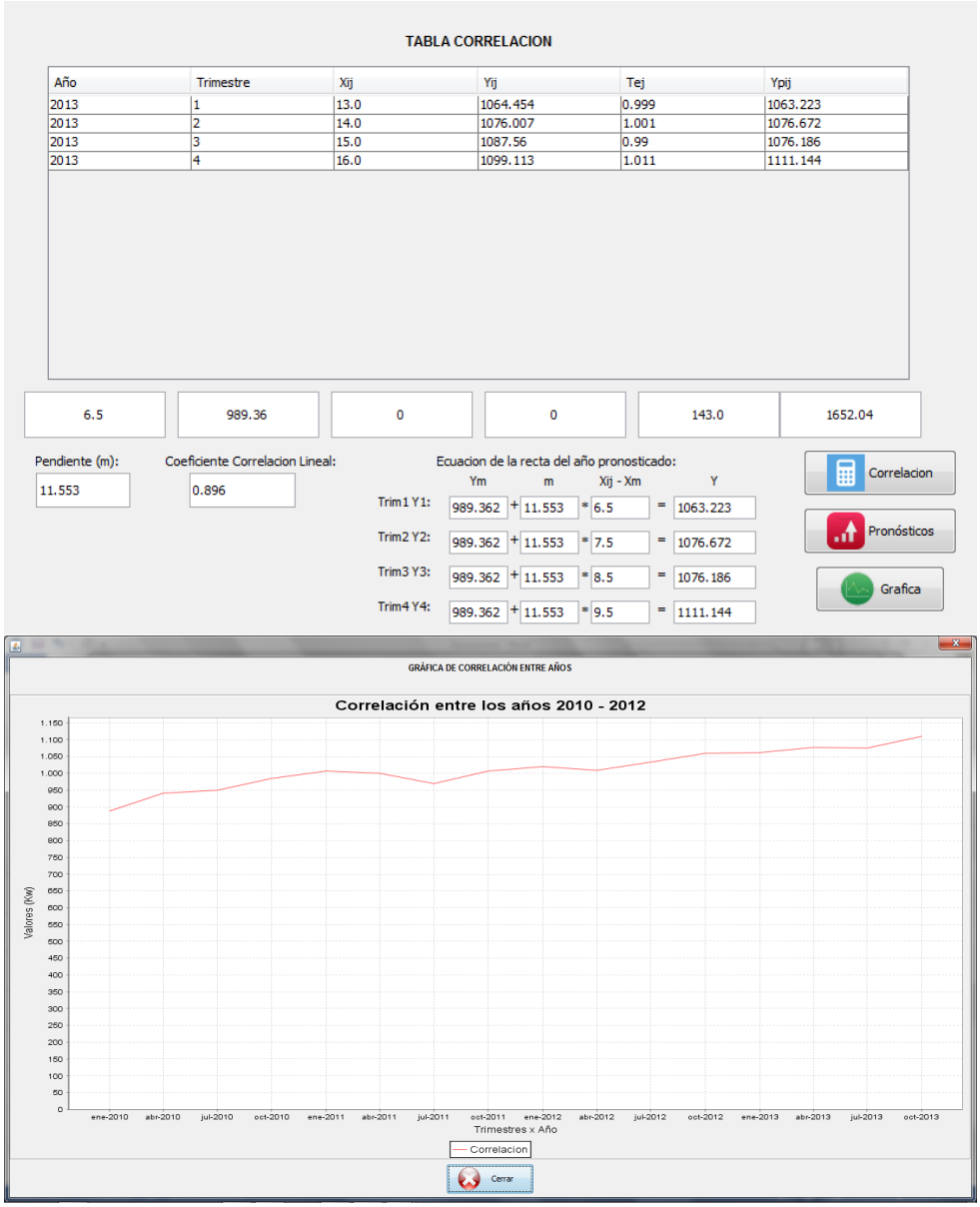


Figura 22. MÉTODO CORRELACIÓN

En la figura 23, se muestra el método de pronóstico.

TABLA PRONÓSTICOS

Xi	Yi'	Yi=ln Yi'	Xi-Xm	Yi-Ym	(Xi-Xm)²	(Xi-Xm)*(Yi-Ym)	(Yi-Ym)²
1.0	838.05	6.73	-5.5	-0.11	30.25	0.63	0.01
2.0	892.03	6.79	-4.5	-0.05	20.25	0.23	0.0
3.0	931.43	6.84	-3.5	-0.01	12.25	0.03	0.0
4.0	932.96	6.84	-2.5	-0.01	6.25	0.02	0.0
5.0	959.82	6.87	-1.5	0.02	2.25	-0.03	0.0
6.0	930.25	6.84	-0.5	-0.01	0.25	0.01	0.0
7.0	935.55	6.84	0.5	-0.0	0.25	-0.0	0.0
8.0	941.49	6.85	1.5	0.0	2.25	0.0	0.0
9.0	972.61	6.88	2.5	0.03	6.25	0.09	0.0
10.0	969.69	6.88	3.5	0.03	12.25	0.11	0.0
11.0	965.76	6.87	4.5	0.03	20.25	0.12	0.0
12.0	1018.91	6.93	5.5	0.08	30.25	0.45	0.01
6.5		6.85	0	0	143.0	1.65	0.03

Pendiente (m): 0.012

Desviación estándar: 1 = 0.025

Raiz: 1.138

C.R.I. 0.26

Resultado: 6.913

Desviación: 1 = 6.78

Pronostico: 13 = 6.921

Varianza estándar: 0.029

Rang. Incert. 0.007

Figura 23. MÉTODO PRONÓSTICO

2.2. Pantallas modulo pronóstico de carga

Caso de uso calcular pronóstico de carga

TABLA V. PRONÓSTICO DE CARGA

NOMBRE DEL CASO DE USO: Calcular pronóstico de carga		**CÓDIGO: 001**
REFERENCIA DE REQUESITOS:	**ACTORES**	**TIPO DE C.U.**
RF001, RF002, RF003, RF004	Administrador	Primario
OBJETIVO(S)	**DESCRIPCIÓN**	
• Escoger la Subestación. • Escoger los alimentadores.	• El administrador escoge la subestación, el alimentador correspondiente, el intervalo del tiempo y presiona el botón listar para cargar los datos en la tabla alimentadores.	
PRE-CONDICIONES	**POST-CONDICIONES**	
• El sistema debe estar ya cargado. • Que el administrador elija el menú principal, la opción [reportes y estadísticas], el submenú [Pronostico]. • El módulo presenta la pantalla Pronóstico de Cargas.	• Permitir a los usuarios ingresar al Módulo de Pronostico de Carga.	
FLUJO NORMAL		

Acción del Administrador	Respuesta del Módulo
1. Escoge el intervalo de tiempo ya sea en Año (JDateChooser), mes (JMonthChooser) o día (JYearChooser).	2. El módulo habilita el tiempo a pronosticar.
3. Escoge la subestación y alimentador a pronosticar.	4. El módulo presenta una lista con las subestaciones.
5. Escoge la potencia activa o reactiva (JComboBox).	6. El módulo presenta una lista con los datos de Potencia Activa (P) y Potencia Reactiva (Q).
7. Escoge el grado de libertad (JComboBox).	8. El módulo presenta una lista con el grado de libertad a escoger.
9. Escoge el porcentaje de certidumbre (JComboBox).	10. El módulo presenta una lista con el nivel de incertidumbre a escoger
11. Presiona el botón listar.	12. El modulo presenta los datos un la tabla de Alimentadores. 13. El caso de uso finaliza.
FLUJO ALTERNO	
Ítem 1: Escoge un intervalo de tiempo campo obligatorio. El caso de uso continúa en el paso 3 del flujo normal de eventos **Ítem 2:** Escoge una subestación campo obligatorio. El caso de uso continúa en el paso 5 del flujo normal de eventos.	

Ítem 3: Escoge un alimentador campo obligatorio. El caso de uso continúa en el paso 5 del flujo normal de eventos.

Caso de uso pronosticar por el método de regresión.

TABLA VI. CALCULAR MÉTODO DE REGRESIÓN

NOMBRE DEL CASO DE USO: Calcular método de regresión		**CÓDIGO: 002**
REFERENCIA DE REQUESITOS:	**ACTORES**	**TIPO DE C.U.**
RF001, RF005, RF007, RF010	Administrador	Primario
OBJETIVO(S)	**DESCRIPCIÓN**	
• Escoger la Subestación. • Escoger los alimentadores.	• El administrador escoge la subestación, el alimentador correspondiente, el intervalo del tiempo y presiona el botón listar para cargar los datos en la tabla alimentadores.	
PRE-CONDICIONES	**POST-CONDICIONES**	
• El sistema debe estar ya cargado. • Que los alimentadores ya estén cargados en el módulo en base al intervalo del tiempo.	• Permitir a los usuarios ingresar al Módulo de Pronostico de Carga.	

FLUJO NORMAL	
Acción del Administrador	**Respuesta del Módulo**
1. Escoge el intervalo de tiempo ya sea en Año (JDateChooser), mes (JMonthChooser) o día (JYearChooser).	2. El módulo habilita el tiempo escogido.
3. Escoge la subestación y los alimentadores correspondientes.	4. El modulo presenta una lista con las subestaciones y alimentadores cargados en el sistema.
5. Escoge en la sección opciones la potencia activa (P) o reactiva (Q), el grado de libertad y porcentaje de certidumbre.	6. El módulo presenta una lista con los datos respectivos a cada opción realizada.
7. Presiona el botón listar.	8. El módulo presenta los datos cargados en una tabla (JTable).
	9. El caso de uso finaliza.
FLUJO ALTERNO	
Ítem 1: Escoge un intervalo de tiempo campo obligatorio. El caso de uso continúa en el paso 3 del flujo normal de eventos **Ítem 2:** Escoge una subestación campo obligatorio. El caso de uso continúa en el paso 5 del flujo normal de eventos.	

Ítem 3: Escoge un alimentador campo obligatorio. El caso de uso continúa en el paso 5 del flujo normal de eventos.

Caso de uso pronosticar por el método de correlación.

TABLA VII. CALCULAR MÉTODO DE CORRELACIÓN

NOMBRE DEL CASO DE USO: Calcular método de correlación		**CÓDIGO: 003**
REFERENCIA DE REQUESITOS:	**ACTORES**	**TIPO DE C.U.**
RF001, RF005, RF007, RF010	Administrador	Primario
OBJETIVO(S)	**DESCRIPCIÓN**	
• Escoger la Subestación. • Escoger los alimentadores.	• El administrador escoge la subestación, el alimentador correspondiente, el intervalo del tiempo y presiona el botón listar para cargar los datos en la tabla alimentadores.	
PRE-CONDICIONES	**POST-CONDICIONES**	
• El sistema debe estar ya cargado. • Que los alimentadores ya estén cargados en el módulo en base al intervalo del tiempo.	• Permitir a los usuarios ingresar al Módulo de Pronostico de Carga.	

FLUJO NORMAL	
Acción del Administrador	**Respuesta del Módulo**
1. Escoge el intervalo de tiempo en Año (JDateChooser). 3. Escoge la subestación y los alimentadores correspondientes. 5. Escoge en la sección opciones la potencia activa (P) o reactiva (Q), el grado de libertad y porcentaje de certidumbre. 7. Presiona el botón listar. 9. Presiona el botón correlación entre valores. 11. Presiona el botón grafica	2. El módulo habilita el tiempo escogido. 4. El modulo presenta una lista con las subestaciones y alimentadores cargados en el sistema. 6. El módulo presenta una lista con los datos respectivos a cada opción realizada. 8. El módulo presenta los datos en la tabla correlación (Jtable). 10. El modulo presenta los datos correspondientes a correlación entre valores e la tabla correlación (JTable). 12. El modulo permite mostrar la gráfica de los años correlacionados 13. El caso de uso finaliza.
FLUJO ALTERNO	

Ítem 1: Escoge un intervalo de tiempo Año campo obligatorio. El caso de uso continúa en el paso 3 del flujo normal de eventos **Ítem 2:** Escoge una subestación campo obligatorio. El caso de uso continúa en el paso 5 del flujo normal de eventos. **Ítem 3:** Escoge un alimentador campo obligatorio. El caso de uso continúa en el paso 5 del flujo normal de eventos.

Caso de uso pronosticar el año siguiente.

TABLA VIII. PRONOSTICAR POR AÑO SIGUIENTE

NOMBRE DEL CASO DE USO: Calcular pronosticar el año siguiente.		**CÓDIGO: 004**
REFERENCIA DE REQUESITOS:	**ACTORES**	**TIPO DE C.U.**
RF001, RF005, RF007, RF010	Administrador	Primario
OBJETIVO(S)	**DESCRIPCIÓN**	
• Escoger la Subestación. • Escoger los alimentadores.	• El administrador escoge la subestación, el alimentador correspondiente, el intervalo del tiempo y presiona el botón listar para cargar los datos en la tabla alimentadores.	
PRE-CONDICIONES	**POST-CONDICIONES**	

• El sistema debe estar ya cargado. • Que los alimentadores ya estén cargados en el módulo en base al intervalo del tiempo.	• Permitir a los usuarios ingresar al Módulo de Pronostico de Carga.
FLUJO NORMAL	
Acción del Administrador	**Respuesta del Módulo**
1. Escoge el intervalo de tiempo en Año (JDateChooser). 3. Escoge la subestación y los alimentadores correspondientes. 5. Escoge en la sección opciones la potencia activa (P) o reactiva (Q), el grado de libertad y porcentaje de certidumbre. 7. Presiona el botón listar.	2. El módulo habilita el tiempo escogido. 4. El modulo presenta una lista con las subestaciones y alimentadores cargados en el sistema. 6. El módulo presenta una lista con los datos respectivos a cada opción realizada. 8. El módulo presenta los datos en la tabla pronóstico (Jtable).
FLUJO ALTERNO	
Ítem 1: Escoge un intervalo de tiempo Año campo obligatorio. El caso de uso continúa en el paso 3 del flujo normal de eventos **Ítem 2:** Escoge una subestación campo obligatorio.	

El caso de uso continúa en el paso 5 del flujo normal de eventos.
Ítem 3: Escoge un alimentador campo obligatorio.
El caso de uso continúa en el paso 5 del flujo normal de eventos.

3. DISEÑO

3.1. DIAGRAMAS DE SECUENCIA.

Diagrama de secuencia pronóstico de carga.

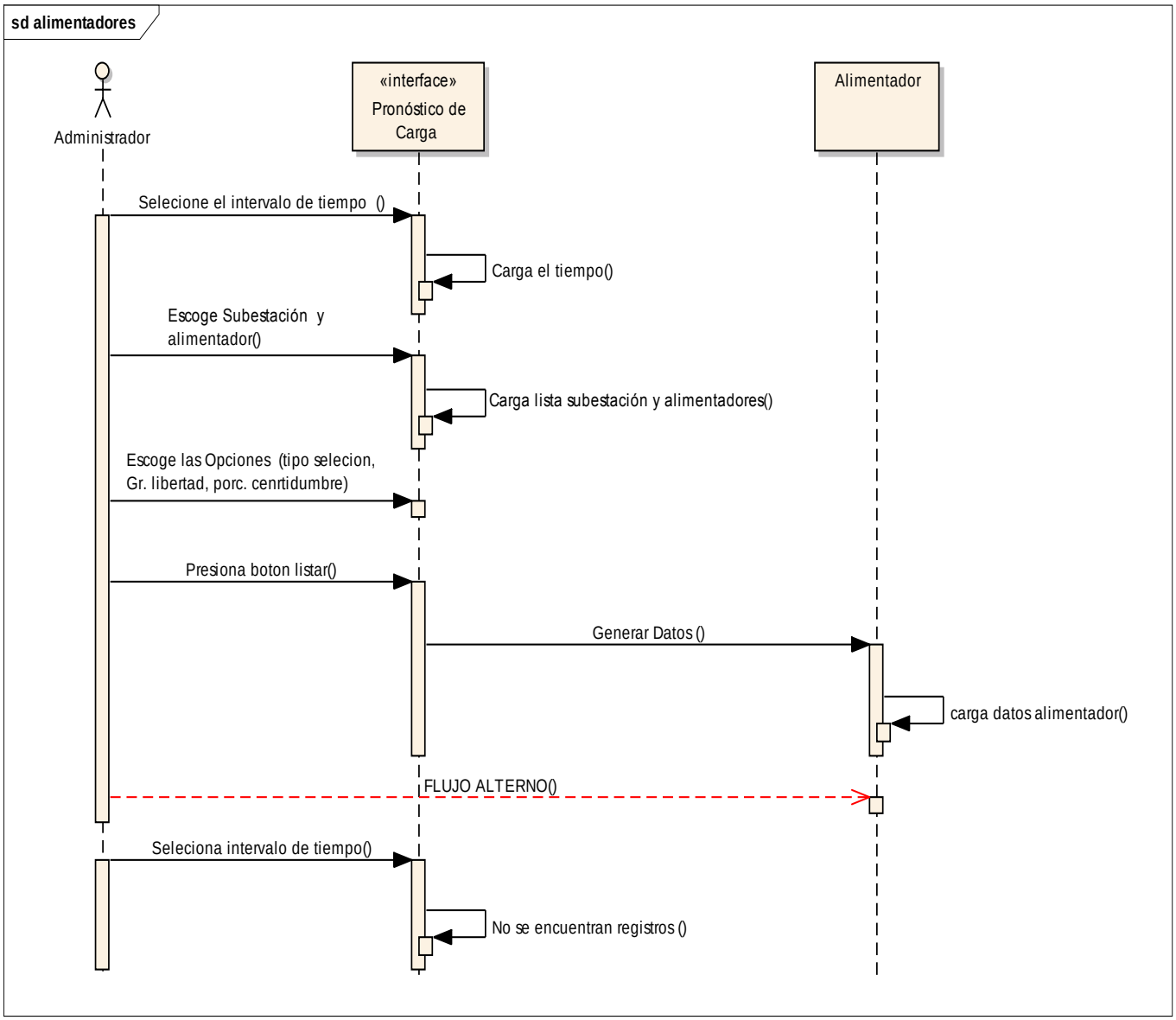


Figura 24. Diagrama de secuencia pronóstico de carga.

Diagrama de secuencia pronóstico por método regresión

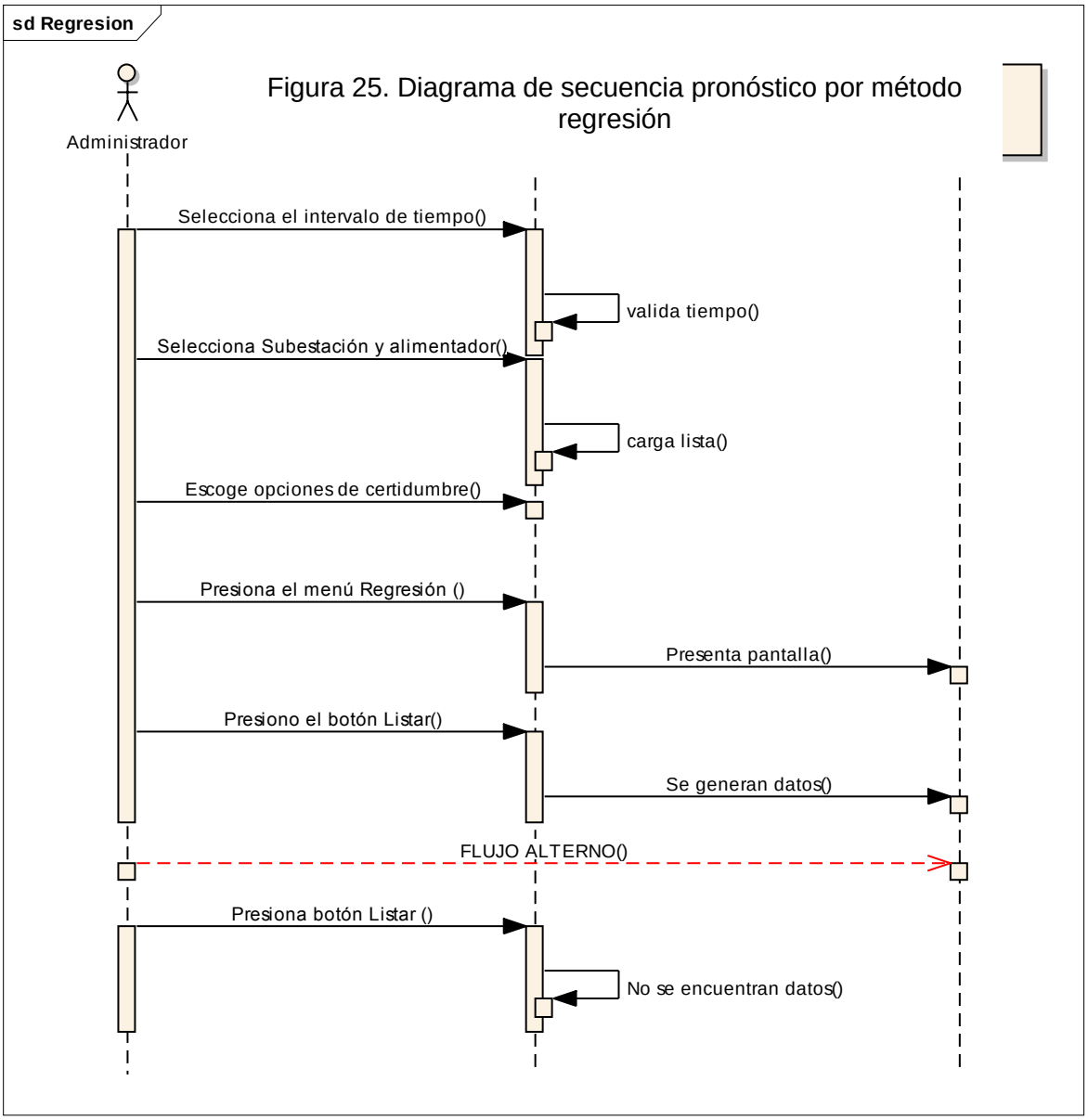


Figura 25. Diagrama de secuencia pronóstico por método regresión

Diagrama de secuencia pronóstico por método de correlación

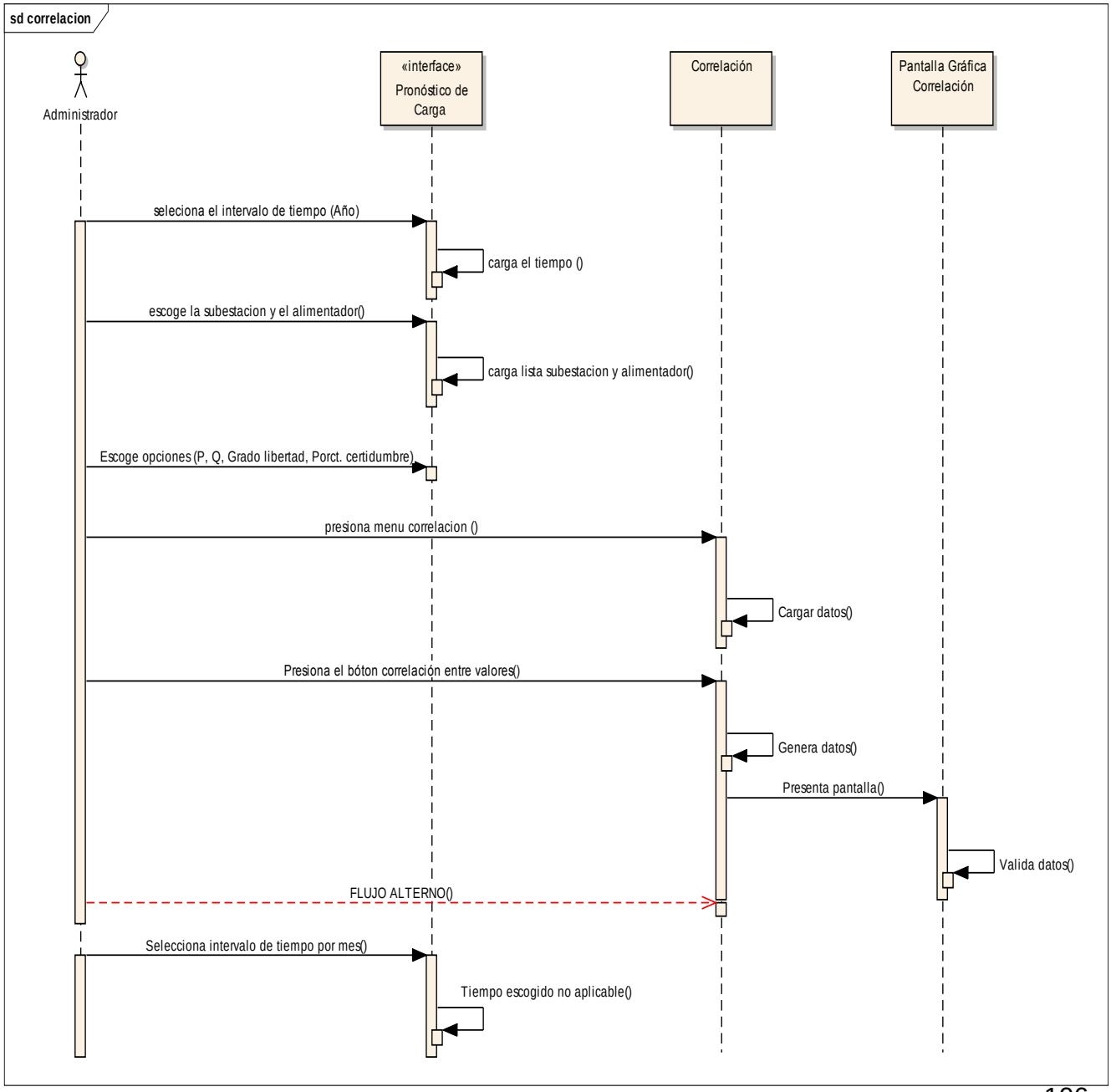


Figura 26.Diagrama de secuencia pronóstico por método correlación

Diagrama de secuencia método por pronóstico

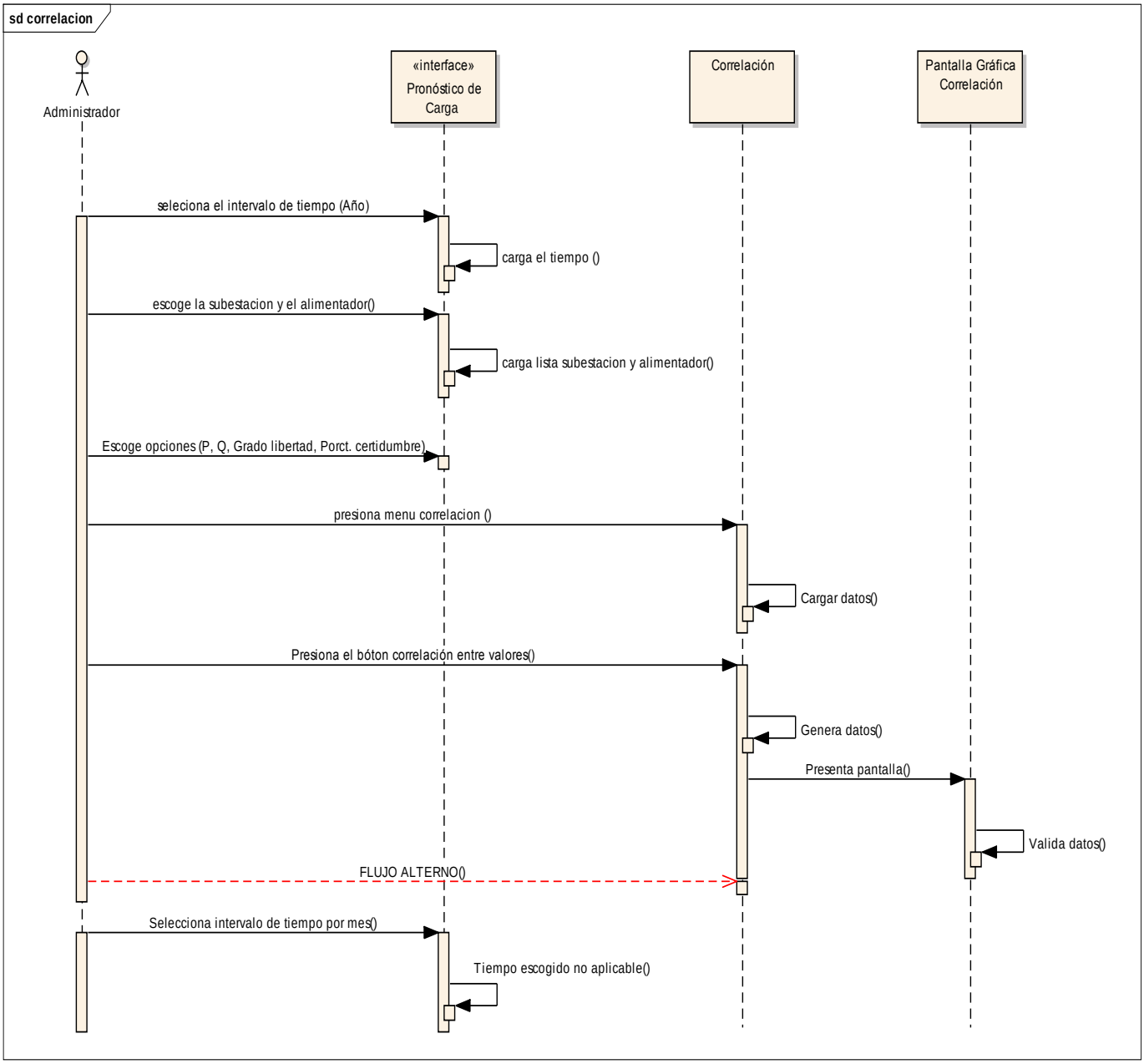


Figura 27. Diagrama de secuencia método por pronóstico

3.2. DIAGRAMAS DE ROBUSTEZ

Diagrama de robustez pronóstico de carga.

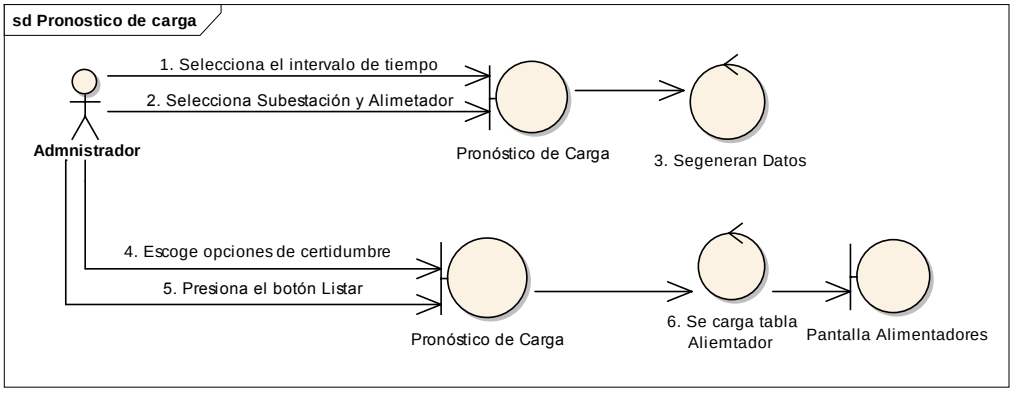


Figura 28. Diagrama de Robustez Pronóstico de Carga

Diagrama de robustez pronóstico por método de regresión.

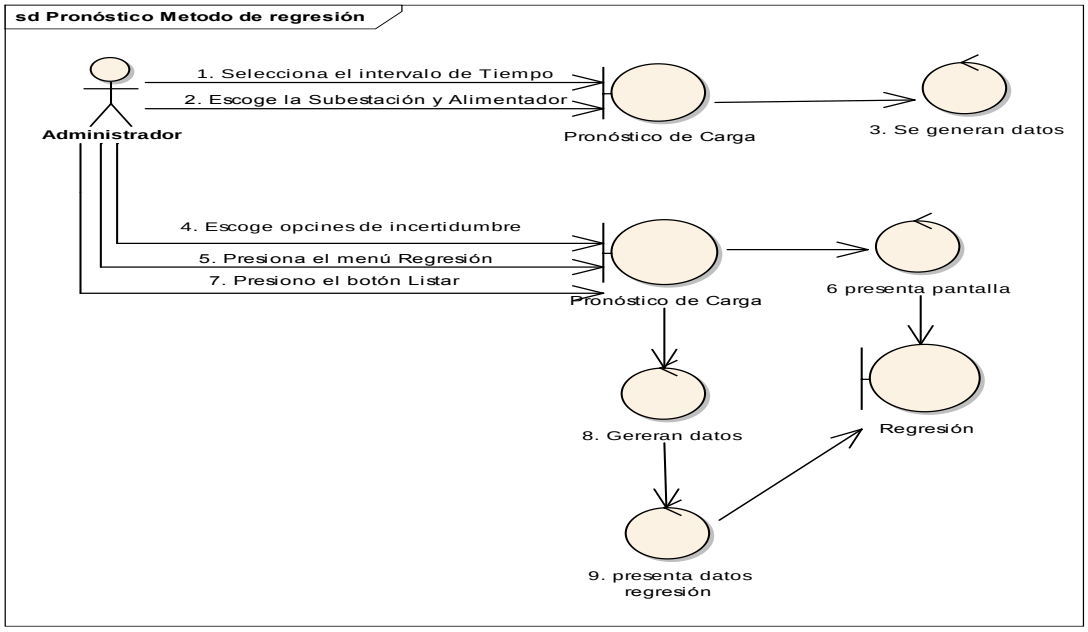


Figura 29. Diagrama de Robustez Pronóstico por Método de Regresión

Diagrama de robustez pronóstico por método de correlación.

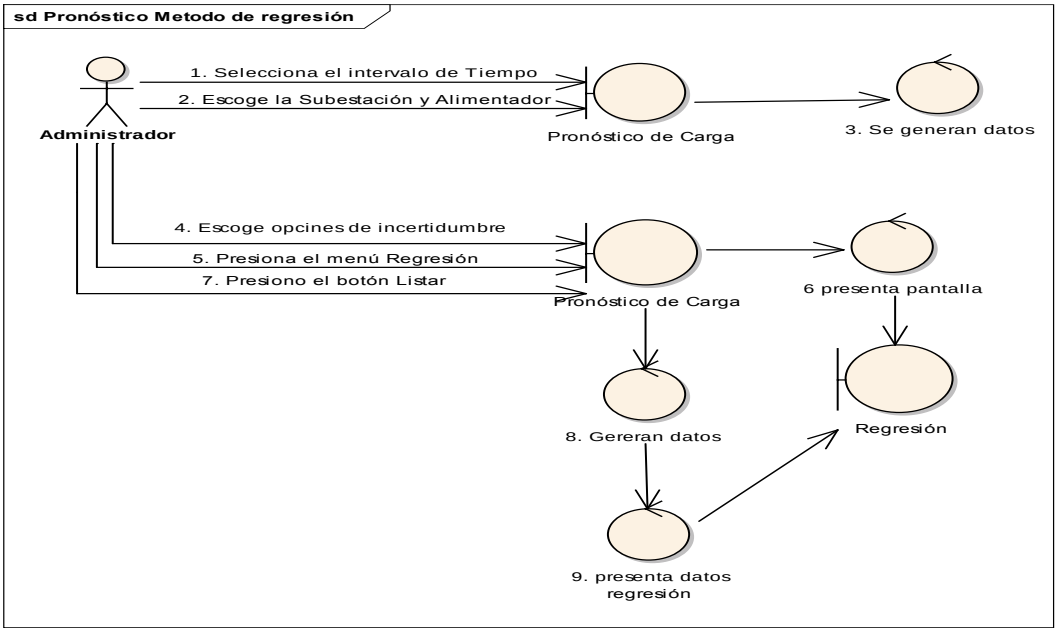


Figura 30. Diagrama de Robustez Pronóstico de Correlación

Diagrama de robustez pronóstico por método de pronóstico.

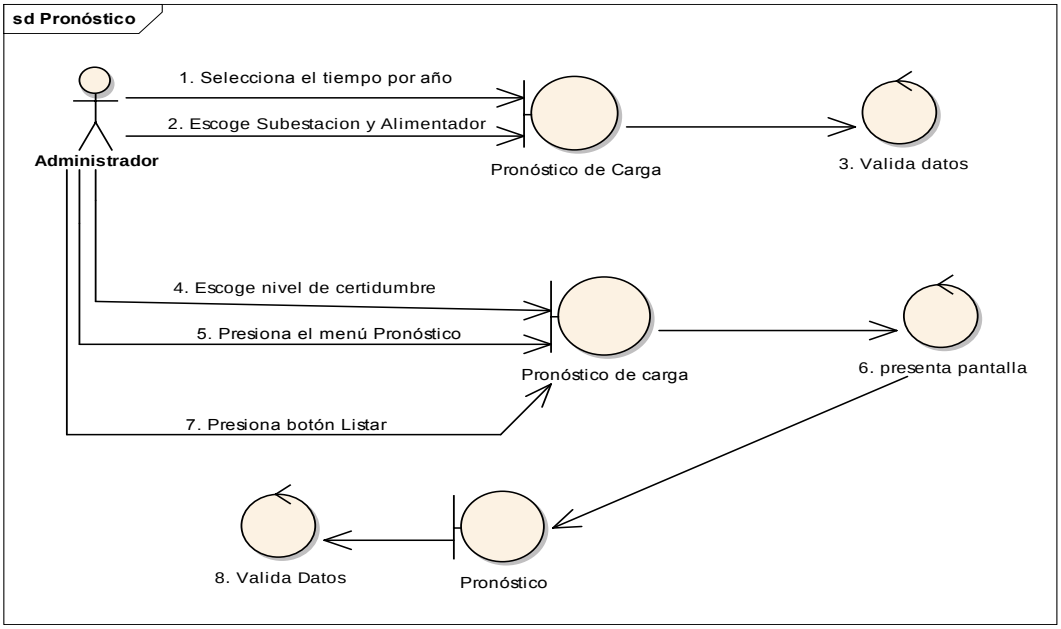


Figura 31. Diagrama de Robustez Pronóstico por Método de Pronóstico

3.3. Diagrama de clases

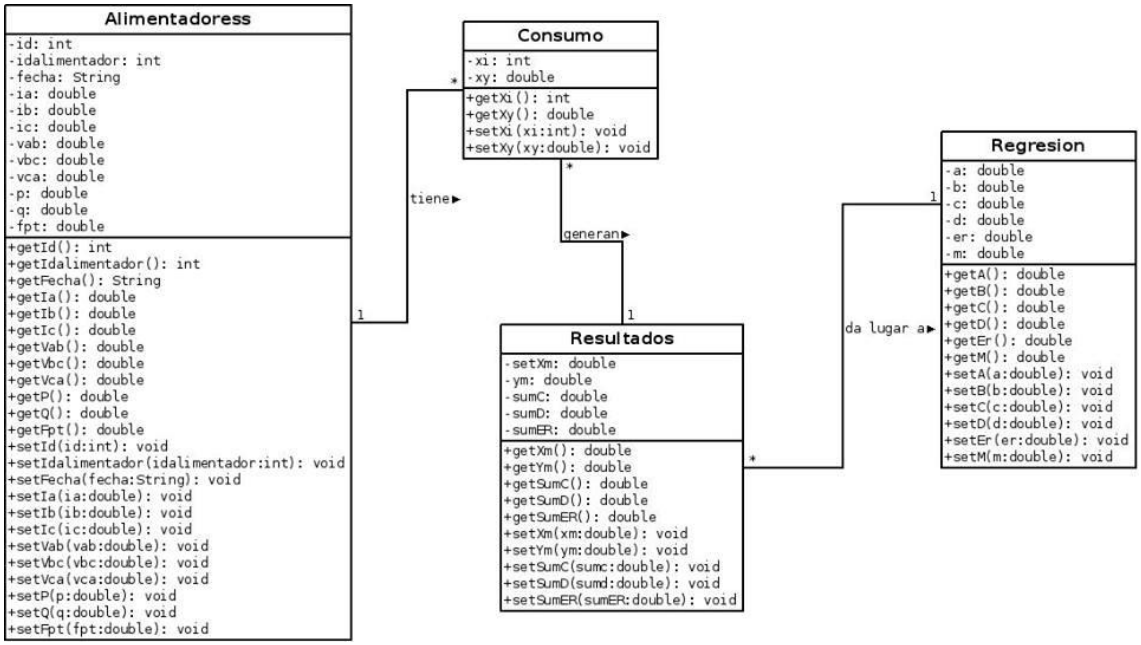


Figura 32. Diagrama de Clases

En la tabla Alimentadoress se encuentran las variables declaradas y los atributos privados de la clase Alimentadoress que se encuentran en la base de datos.

En la tabla Consumo, se encuentran los atributos get y set de la clase Consumo y sus variables declaradas para los valores ingresados en las tablas de cada método.

En la tabla Resultados se encuentran declaradas las variables de los resultados obtenidos en la clase_Resultados, y sus atributos los mismos que le permitirán acceder a esta clase.

En la tabla Regresiones se encuentran declaradas las variables de los métodos de regresión, correlación y pronóstico de la clase_Regresiones y los atributos que le permiten acceder a la misma.

3.4. Diagrama De Base De Datos

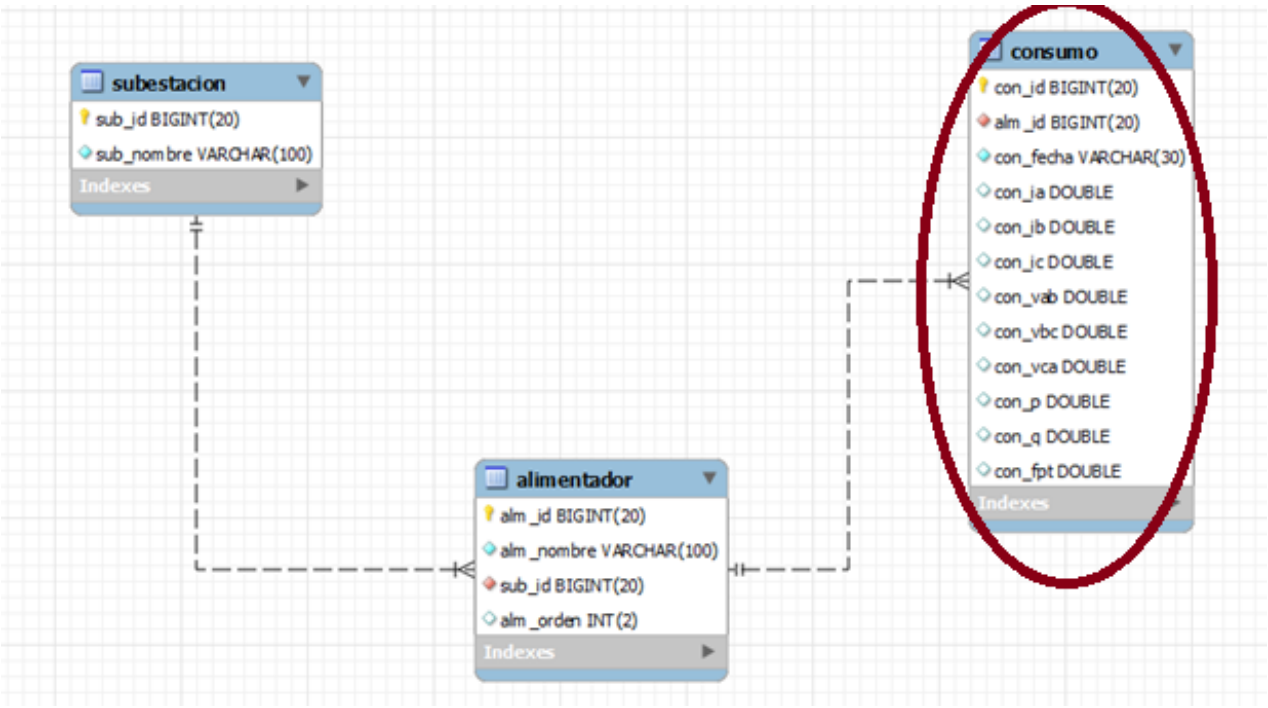


Figura 33. Diagrama de Base de Datos

En el diagrama de Base de Datos vemos que está integrado por tres tablas: la tabla de subestación, la tabla de alimentador y la tabla de consumo; para nuestro modulo haremos uso únicamente de la tabla consumo ya que aquí se alojan los valores que se utilizaran para realizar los cálculos en los métodos de regresión, correlación y pronostico en sus intervalos de día-mes-año.

3.5. Resultados de la Aplicación

En el módulo denominado “DISEÑO Y CREACIÓN DEL MÓDULO DE PRONÓSTICO DE CARGA, PARA EL DESARROLLO DE UN SOFTWARE PARA EL ESTUDIO DE REDES DE DISTRIBUCIÓN ELÉCTRICA DE LA CIUDAD DE LOJA”, procedemos a:

1. Se selecciona el método a calcular, luego dependiendo del mismo se selecciona el tipo de intervalo a pronosticar, ya sea en día-mes-año.
2. Al seleccionar el método de regresión procedemos a escoger el intervalo en día-mes-año, luego mandamos a listar los valores, se cargan los valores en cada uno de los campos asignados y se procede a dar el pronóstico dependiendo del intervalo escogido.
3. Si seleccionamos el método de correlación tan solo nos permitirá calcularlo en el intervalo de año ya que este método es exclusivo para este tipo de pronóstico, entonces se escoge los años a correlacionarse y se manda a listar, se llenaran cada uno de los campos asignados en la tabla y se mostrara el cálculo de la pendiente, la ecuación de la recta, la correlación de años, el pronóstico futuro y así mismo se permitir hacer la gráfica correspondiente de los años correlacionados y del año pronosticado.
4. Si escogemos el método de pronostico este tan solo se podrá calcular en el intervalo de año, es decir se selecciona el año y se manda a listar, luego se llenan los campos vacíos de la tabla y se pronostica el primer mes del siguiente año.

4. IMPLEMENTACIÓN

4.1 Método de Operaciones Base de Datos

Aquí se muestra cada una de las sentencias utilizadas para llamar los alimentadores respectivos de las subestaciones y realizar los cálculos predeterminados en el intervalo de día-mes-año, o tan solo de los años según corresponda el método.

```
public static List getListAlimAll(int idAlim){
        entidad.getTransaction().begin();
        List lista = null;
        try {
            lista = entidad.createNativeQuery("call subestaciones.get_alimentadores("+idAlim+")").getResultList();  // devuelve List<Object[]>
            entidad.getTransaction().commit();

        } catch (Exception e) {
            entidad.getTransaction().rollback();
        }
        return lista;
    }

    public static List getListAlimByDate(String date, int idAlim){
        entidad.getTransaction().begin();
        List lista = null;
        try {
            lista = entidad.createNativeQuery("call subestaciones.get_alimentadores_hour_day('"+date+"','"+idAlim+")").getResultList();  // devuelve List<Object[]>
            entidad.getTransaction().commit();

        } catch (Exception e) {
            entidad.getTransaction().rollback();
        }
        return lista;
    }
```

```
    public static List getListAlimByDayMonth(int mes, int anio, int idAlim){
        entidad.getTransaction().begin();
        List lista = null;
        try {
            lista = entidad.createNativeQuery("call
subestaciones.get_alimentadores_day_month("+mes+","+anio+","+idAlim+")").getResult
List(); // devuelve List<Object[]>
            entidad.getTransaction().commit();

        } catch (Exception e) {
            entidad.getTransaction().rollback();
        }
        return lista;
    }

    public static List getListAlimByMonthYear(int anio, int idAlim){
        entidad.getTransaction().begin();
        List lista = null;
        try {
            lista = entidad.createNativeQuery("call
subestaciones.get_alimentadores_month_year("+anio+","+idAlim+")").getResultList();   //
devuelve List<Object[]>
            entidad.getTransaction().commit();

        } catch (Exception e) {
            entidad.getTransaction().rollback();
        }
        return lista;
    }
```

4.2. Método de Regresión

En este método se procederá a realizar los cálculos en un intervalo de día-mes-año, dependiendo de la necesidad del administrador, en donde se debe selecciona la subestación y posteriormente el alimentador a calcular.

```
private void calcularRegresion() {
        calcXmYm();
```

```
        regresiones = new ArrayList<>();
        double sc = 0, sd = 0, ser = 0;
        for (Consumo c : consumos) {
            Regresion r = new Regresion();
            r.setA(c.getXi() - resReg.getXm());
            r.setB(c.getYi() - resReg.getYm());
            r.setC(Math.pow(r.getA(), 2));
            r.setD(r.getA() * r.getB());
            r.setEr(Math.pow(r.getB(), 2));
            r.setM(r.getD() / r.getC());
            regresiones.add(r);
            sc = sc + r.getC();
            sd = sd + r.getD();
            ser = ser + r.getEr();
        }
        resReg.setSumC(sc);
        resReg.setSumD(sd);
        resReg.setSumER(ser);
        resReg.setM(resReg.getSumD() / resReg.getSumC());
        calcDesvReg();
        calcDesvEstReg();
        calcYpYtReg();
        calcPronosReg();
    }
```

4.3. Método de Correlación

En este método se procede a seleccionar dos años entre los cuales se va hacer la correlación de sus datos.

```
private void calcularRegCorr() {
        calcXmYm();
        regresiones = new ArrayList<>();
        double sc = 0, sd = 0, ser = 0;
        for (Consumo c : consumos) {
            Regresion r = new Regresion();
            r.setA(c.getXi() - resReg.getXm());
            r.setB(c.getYi() - resReg.getYm());
```

```
            r.setC(Math.pow(r.getA(), 2));
            r.setD(r.getA() * r.getB());
            r.setEr(0);
            r.setM(r.getD() / r.getC());
            regresiones.add(r);
            sc = sc + r.getC();
            sd = sd + r.getD();
            ser = ser + r.getEr();
        }
        resReg.setSumC(sc);
        resReg.setSumD(sd);
        resReg.setSumER(ser);
        resReg.setM(resReg.getSumD() / resReg.getSumC());
//        calcCorrVal();
    }
```

Dentro de este método podemos graficar los datos de los años correlacionados, para esto se usa el siguiente código.

```
private void graficar() {
        double trim1 = 0;
        double trim2 = 0;
        double trim3 = 0;
        double trim4 = 0;
        DefaultCategoryDataset dataset = new DefaultCategoryDataset();
        // yij, trimestre, año
        int c = 0;
        int n = 0;
        for (double[] a : ijs) {
            for (int i = 0; i < a.length; i++) {
                if (i == 0) {
                    trim1 += a[i];
                } else if (i == 1) {
                    trim2 += a[i];
                } else if (i == 2) {
                    trim3 += a[i];
                } else if (i == 3) {
                    trim4 += a[i];
                }
```

```
            dataset.setValue(consumos.get(c).getYi(), "Trimestre "+(i+1), "Año "+(auxAnioInic+n));
            c++;
        }
        n++;
    }

    double tej_trim1 = trim1/ijs.size();
    double tej_trim2 = trim2/ijs.size();
    double tej_trim3 = trim3/ijs.size();
    double tej_trim4 = trim4/ijs.size();

    double yij_trim1 = resReg.getYm() + resReg.getM()*((consumos.size()+1)-resReg.getXm());
    double yij_trim2 = resReg.getYm() + resReg.getM()*((consumos.size()+2)-resReg.getXm());
    double yij_trim3 = resReg.getYm() + resReg.getM()*((consumos.size()+3)-resReg.getXm());
    double yij_trim4 = resReg.getYm() + resReg.getM()*((consumos.size()+4)-resReg.getXm());

    double ypij_trim1 = yij_trim1 * tej_trim1;
    double ypij_trim2 = yij_trim2 * tej_trim2;
    double ypij_trim3 = yij_trim3 * tej_trim3;
    double ypij_trim4 = yij_trim4 * tej_trim4;

    dataset.setValue(ypij_trim1, "Trimestre 1", "Año "+(auxAnioInic+n));
    dataset.setValue(ypij_trim2, "Trimestre 2", "Año "+(auxAnioInic+n));
    dataset.setValue(ypij_trim3, "Trimestre 3", "Año "+(auxAnioInic+n));
    dataset.setValue(ypij_trim4, "Trimestre 4", "Año "+(auxAnioInic+n));

    JFreeChart chart = ChartFactory.createBarChart("Correlación entre años",
        "Años", "Valores", dataset, PlotOrientation.VERTICAL, true,true, false);
    chart.setBackgroundPaint(Color.white);
    chart.getTitle().setPaint(Color.black);
    CategoryPlot p = chart.getCategoryPlot();
    p.setRangeGridlinePaint(Color.red);
    // Mostrar Grafico
    ChartPanel chartPanel = new ChartPanel(chart);
    DialogGrafico dg = new DialogGrafico(null, true, chartPanel);
```

```
        dg.setVisible(true);
    }
```

4.4. Método de Pronóstico

Este método permite realizar el pronóstico para el siguiente año, es decir se carga los valores de los dos últimos años y se pronostica el siguiente que aún está empezando.

```
private void listPronosticos() {
        consumos = new ArrayList<>();
        aux = new ArrayList<>();
        ArrayList<Object[]> ae = null;
        resReg = new Resultados();
        if (tipoBusq == -1) {
            ae = (ArrayList<Object[]>) OperacionesBD.getListRegByYears(idAlim,
valOpc);
        } else if (tipoBusq == 0) {
            SimpleDateFormat sdf = new SimpleDateFormat("YYYY-MM-dd");
            String fecha = sdf.format(form.getDchDia().getDate());
            ae = (ArrayList<Object[]>) OperacionesBD.getListRegByDate(fecha, idAlim,
valOpc);
        } else if (tipoBusq == 1) {
            int anio = form.getDchMesAnio().getYear();
            int mes = form.getDchMes().getMonth();
            ae = (ArrayList<Object[]>) OperacionesBD.getListRegByDayMonth(mes + 1,
anio, idAlim, valOpc);
        } else if (tipoBusq == 2) {
            int anio = form.getDchAnio().getYear();
            ae = (ArrayList<Object[]>) OperacionesBD.getListRegByMonthYear(anio,
idAlim, valOpc);
        }
        if (ae != null && !ae.isEmpty()) {
            int c = 1;
            for (Object[] o : ae) {
                aux.add((double) o[1]);
                Consumo cs = new Consumo(c, Math.log((double) o[1]));
                consumos.add(cs);
                c++;
            }
```

```
        form.getSpYPron().setModel(new
SpinnerNumberModel(consumos.get(0).getXi(),                consumos.get(0).getXi(),
consumos.size(), 1));
        form.getSpNPron().setModel(new
SpinnerNumberModel(consumos.get(0).getXi(),                consumos.get(0).getXi(),
consumos.size(), 1));
        form.getSpXpPron().setModel(new  SpinnerNumberModel(consumos.size()  +
1, consumos.size() + 1, consumos.size() * 2, 1));
        calcularRegresionPron();
        modelTblPronosticos();
        llenaCamposPron();
    } else {
        quitaModeloTbl(form.getTblProno());
    }
}
```

4.5. Cálculo Grados de Libertad y Rango de Certidumbre

Este código permite realizar el cálculo de los grados de libertad dados por la tabla de Student en donde la probabilidad que para nuestros casos será del 99.95% de certidumbre.

```
public static String[] getGradLib(){
    String[] gl = new String[tblStudent.length-1];
    for (int i = 1; i < tblStudent.length; i++) {
        gl[i-1] =  String.valueOf((int)tblStudent[i][0]);
    }
    return gl;
}

public static String[] getPorcCert(){
    String[]pc = new String[tblStudent[0].length-1];
    for (int i = 1; i < tblStudent[0].length; i++) {
        double val = (1-tblStudent[0][i])*100;
        pc[i-1] = String.valueOf(val);
    }
    return pc;
}
```

5. PRUEBAS UNITARIAS

Una prueba unitaria es “una forma de probar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado.”.

Para algunos, este tipo de pruebas es una pérdida de tiempo o una inutilidad. Hay que decir que crear las pruebas supone un tiempo adicional al que dedicamos a programar la aplicación en sí.

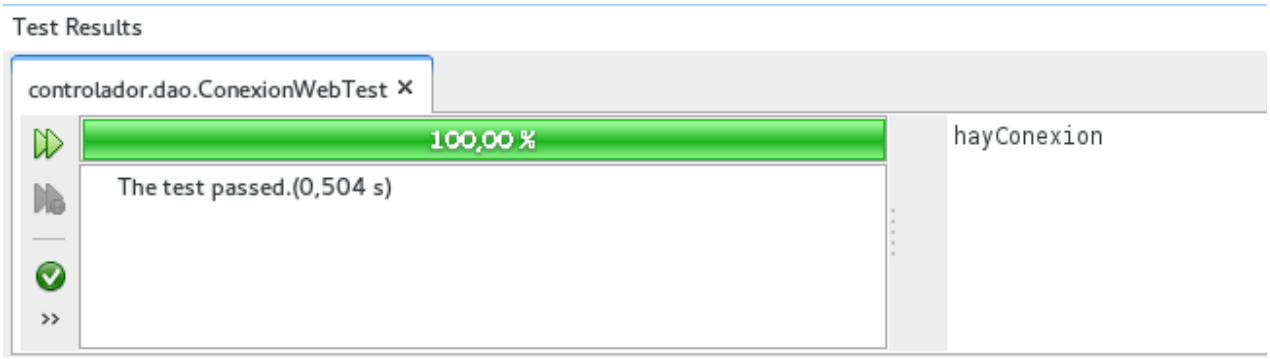
Debemos valorar si merece la pena o no, según las dimensiones y la complejidad de nuestro proyecto.

Lo cierto es que si las hacemos, dependiendo del caso, podemos ahorrarnos mucho tiempo, ya que habremos automatizado las pruebas e instantáneamente sabremos si nuestra aplicación cumple con los resultados correctos.

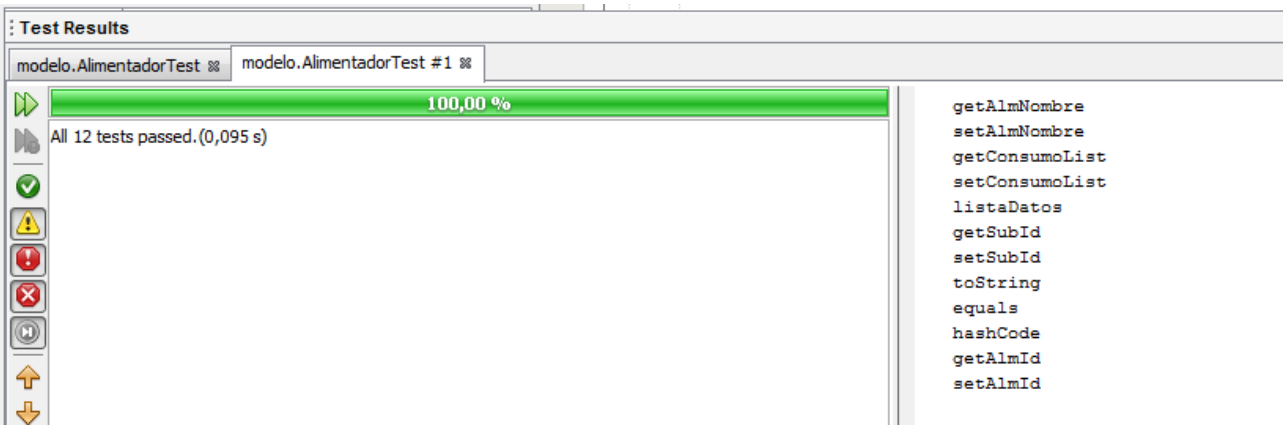
Mejoraremos la calidad del software, asegurándonos de que aunque hagamos cambios en nuestro código, cumpliremos con los requisitos esperados.

En un futuro podremos modificar, ampliar o eliminar código y las pruebas unitarias seguirán sirviéndonos para realizar las comprobaciones pertinentes.

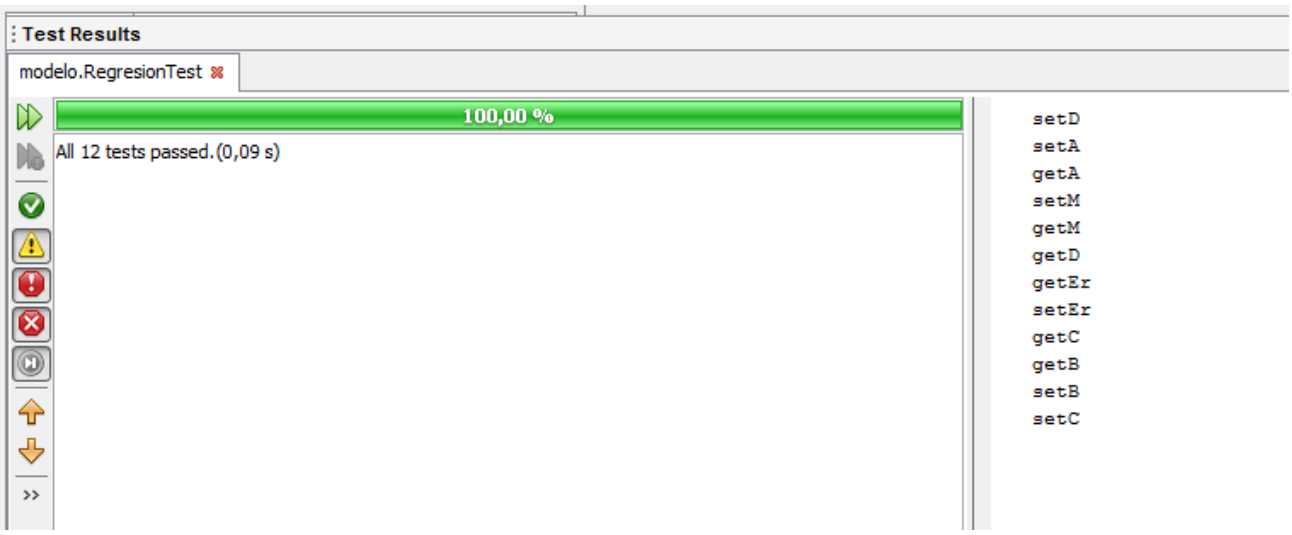
Se realizó los test a las clases principales como ejemplo se muestran los resultados de la ejecución del test conexión web, se puede notar que en este caso el test estuvo correcto pues no había respuesta en el puerto 8080.



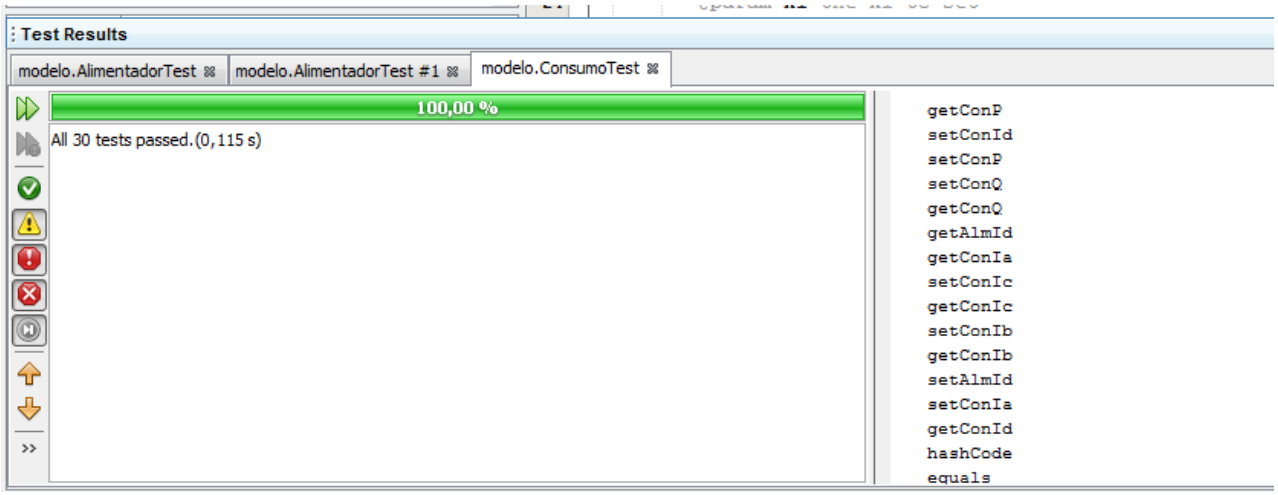
Se realizó los test a la clase alimentadores, en donde este funcionó en su totalidad dando como resultado que todos sus valores están correctamente almacenados.



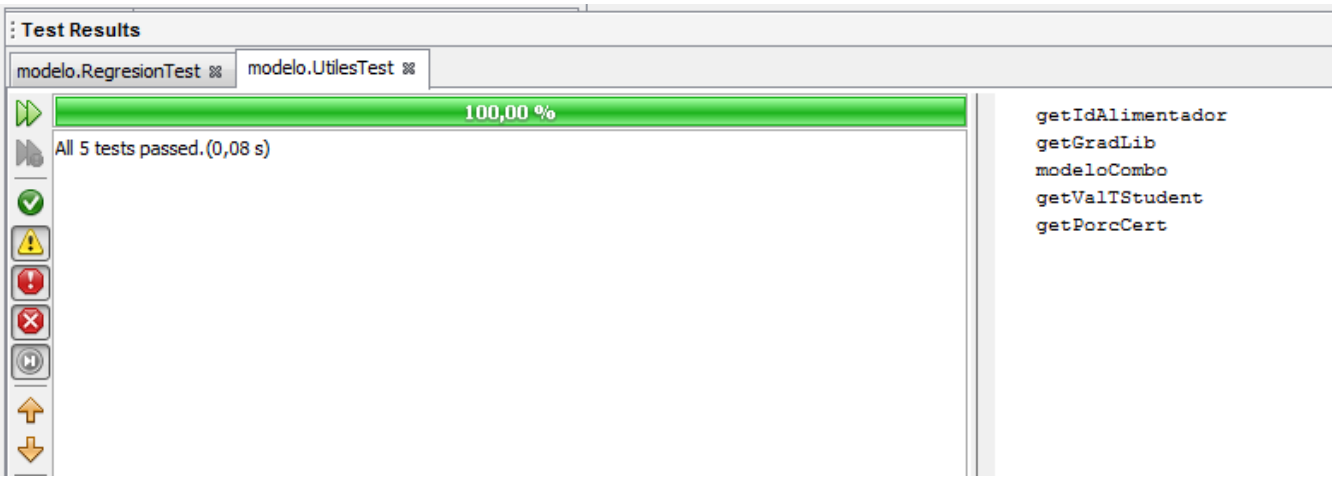
En la ejecución del test de la clase regresión se observa que sus datos están correctamente aplicados, y se ejecuta al 100% la prueba.



Se aplica el test a la clase resultados en donde arroja los parámetros y sus variables, dando una ejecución del 100% sin margen de error.



Y por último se aplica el test a la clase útiles, en donde se hace la llamada a las funciones ejecutadas dentro de las operaciones de cada método y da como resultado una ejecución satisfactoria del 100%.



g. DISCUSIÓN

1. Desarrollo De La Propuesta Alternativa

El presente proyecto de tesis denominado **"DISEÑO Y CREACIÓN DEL MÓDULO DE PRONÓSTICO DE CARGA, PARA EL DESARROLLO DE UN SOFTWARE PARA EL ESTUDIO DE REDES DE DISTRIBUCIÓN ELÉCTRICA DE LA CIUDAD DE LOJA",** dio como resultado la construcción del módulo, desarrollado con el lenguaje de programación JAVA haciendo uso del entorno de programación NetBeans, MySQL como gestor de base de datos y One Server como servidor web.

El objetivo general así como los objetivos específicos fueron englobados y alcanzados en su totalidad, gracias al módulo ordenado de métodos y técnicas orientadas al desarrollo de la investigación y a la metodología de desarrollo de software ICONIX. A continuación evaluaremos los objetivos que fueron planteados en el proyecto de tesis:

Objetivo Especifico 1: Definir el lenguaje de programación que este afín a las necesidades del proyecto.

En esta etapa se analizó cada uno de los lenguajes de programación teniendo en cuenta las necesidades del proyecto y analizando las mejores opciones para el desarrollo rápido, eficiente y objetivo del mismo, decidiéndose por el uso de JAVA con su entorno de programación NetBeans.

Objetivo Especifico 2: Trabajar con las normativas para la evaluación de redes eléctricas estipuladas por la EERSSA y el CONELEC.

Se realizó un estudio minucioso sobre estas normas para poder aplicarlas y así cumplir con los estatutos y normas que lleva la EERSSA.

Objetivo Especifico 3: Determinar los requisitos funcionales y no funcionales de la empresa.

Se determinó cada uno de los requerimientos de la empresa a través de charlas con el Ing. Jorge Carrión González, donde se identificó actores y necesidades de la EERSSA.

Objetivo Especifico 4: Desarrollar un diagrama de clases de alto nivel definido como modelo de dominio.

En esta etapa se identificó cada uno de las clases del módulo con sus atributos y métodos.

Objetivo Especifico 5: Determinar los casos de uso del software.

Se procedió a determinar cada uno de los actores, requisitos y objetivos del módulo.

Objetivo Especifico 6: Realizar el análisis y diseño preliminar del software: describir los casos de uso, diagramas de robustez y diagrama de clases.

Con el diseño de los casos de uso luego procedemos a identificar las actividades que desempeña cada actor dentro del módulo, así mismo se realizó el diseño de los diagramas de robustez y clases.

Objetivo Especifico 7: Programar los métodos y modelos (algoritmo) para el pronóstico de carga.

Se realizó la codificación de los métodos del algoritmo.

Objetivo Especifico 8: Diseñar la interfaz gráfica de usuario en base a los métodos y modelos (algoritmos) del sistema.

Se diseñó una interfaz gráfica amigable con el usuario.

Objetivo Especifico 9: Realizar las pruebas necesarias para determinar que el software está acorde a los requerimientos de la EERSSA.

En esta etapa se realizó pruebas de caja blanca y caja negra al módulo para ver la funcionalidad del mismo y así corregir errores o vacíos en el mismo, con el Ing. Jorge Carrión encargado del proyecto y representante de la EERSSA.

2. Valoración técnica económica ambiental

En cuanto al aspecto económico, los recursos materiales fueron adquiridos por el tesista, en su totalidad.

Por lo antes manifestado, la ejecución del proyecto fue totalmente factible y dando cumplimiento a todos los objetivos planteados.

Los recursos utilizados se detallan a continuación:

- **Recursos Humanos**

Para el desarrollo del proyecto se ha necesitado profesionales de dos perfiles, el director de tesis, quien nos ayudó en la dirección del proyecto siguiendo las guías institucionales; y, un asesor con conocimiento en Redes de distribución eléctrica, que nos orientó en la primera etapa del módulo.

Debido al carácter investigativo y académico del proyecto, este se llevó a cabo por un egresado de la Carrera de Ingeniería en Sistemas, que hizo las veces de analista y programador.

Por tanto, para calcular el coste de personal se ha tenido en cuenta dos perfiles: el asesor con conocimientos en Redes de distribución eléctrica y el de analista/programador.

La tabla IX refleja el número de horas que han empleado cada uno de estos perfiles en el desarrollo del proyecto. Este número de horas de trabajo multiplicado por el coste por hora de cada uno, da como resultado el coste total del recurso humano.

TABLA IX. RECURSOS HUMANOS

TALENTO HUMANO			
EQUIPO TRABAJO	**TIEMPO (HORAS)**	**PRECIO/ HORA ($)**	**VALOR TOTAL**
Carrión González Soraya Yolanda	640	2.50	$1600.00
Asesor 1	240	7.50	$1800.00
Asesor 2	240	7.50	$1800.00
SUBTOTAL			$5200.00

- **Recursos Materiales**

Para el desarrollo del proyecto ha sido necesario el uso de material de oficina empleado para la toma de notas auxiliares, almacenamiento de la información y documentación final.

Como resultado se presenta la tabla X que indica los costes de material desglosados:

TABLA X. RECURSOS MATERIALES

RECURSOS MATERIALES			
RUBRO	**CANT.**	**UNIDAD**	**VALOR TOTAL**
Hojas A4	280	0.03	$8.40
Anillado	5	1.50	$7.50
Materiales de oficina	varios	Varios	$60.00
Perfil	8	0.50	$4.00
CD's	6	0.45	$2.70
Dvd's	4	0.50	$2.00
Empastados	4	5,00	$20,00
SUBTOTAL			**$ 104,60**

- ## Recursos Técnicos/Tecnológicos

Estos recursos fueron divididos en tres secciones: Recursos de Hardware, software y Comunicaciones, a continuación los detallamos:

Recursos de Hardware

El uso de equipos es parte de los materiales usados en el desarrollo del presente proyecto, los cuales son indicados en la tabla XI:

TABLA XI. RECURSOS DE HARDWARE

RECURSOS FÍSICOS

RUBROS	CANT.	VALOR	TIEMPO UTILIZADO (M)	PRECIO /MES	VALOR TOTAL
Portátil HP Pavilion dv5	1	$ 1700.00	12	$ 28.33	$ 340.00
Impresora	1	$ 75.00	12	$ 1.25	$15.00
SUBTOTAL					$ 355.00

Recursos de Software

El software que usamos para el desarrollo de la aplicación es libre, por tal razón no se desembolsó ninguna cantidad de dinero. En la tabla XII se detalla el software utilizado:

TABLA XII. RECURSOS DE SOFTWARE

RECURSOS SOFTWARE			
RUBRO	**CANT.**	**UNIDAD**	**VALOR TOTAL**
Sistemas Operativo Windows 7	1	0.00	$0.00
Herramienta de Programación de java (NetBeans 8.0)	1	0.00	$0.00
Paquete de Ofimática Office	1	0.00	$0.00
MySQL-Workbench 6.1	1	0.00	$0.00
Herramientas de Diseño (Enterprise Architect)	1	135.00	$135.00
RECURSOS SOFTWARE EN LÍNEA			
Google Académico	N	0.00	$0.00
SUBTOTAL			**$ 135.00**

Recursos de Comunicación

El uso de internet y la adquisición del dominio, son parte de este recurso, que fueron imprescindibles para poder realizar las consultas. En la tabla XIII se detalla el costo por cada uno:

TABLA XIII. RECURSOS DE COMUNICACIÓN

RECURSOS DE COMUNICACIÓN	
RUBRO	**COSTO**
Recursos de Internet (Facebook, Twiter, skype, correo electrónico, Drop box)	$ 340.00
Llamadas Telefónicas (fijo, móvil)	$ 60.00
SUBTOTAL:	**$ 400.00**

Resumen del Presupuesto

En la tabla XIV se resume el balance del coste total del proyecto de tesis, para ello se suman los costes obtenidos totalmente.

TABLA XIV. RESUMEN DEL PRESUPUESTO

COSTO TOTAL	
RECURSOS	**COSTOS**
TALENTO HUMANO	$ 5200.00
BIENES	$ 355.00
SERVICIO	$ 720.00

IMPREVISTO	$ 627.50
TOTAL:	**$ 6902.50**

El proyecto tiene un coste total de **$6902,50**

h. CONCLUSIONES

- La utilización de la Metodología ICONIX fue la adecuada para el desarrollo del módulo, por cuanto facilito realizar la captura de requisitos, el desarrollo del análisis y diseño del módulo, permitiendo calcular los datos que se debe obtener en cada tramo.
- Seguir las normativas del CONELEC y la EERSSA permitió trabajar ordenadamente al momento de definir el tipo de valores que se va a utilizar y manejar dentro de la base de datos
- El definir los requisitos funcionales y no funcionales permitió determinar las necesidades del usuario y los casos de uso.
- La creación del módulo de "DISEÑO Y CREACIÓN DEL MÓDULO DE PRONÓSTICO DE CARGA, PARA EL DESARROLLO DE UN SOFTWARE PARA EL ESTUDIO DE REDES DE DISTRIBUCIÓN ELÉCTRICA DE LA CIUDAD DE LOJA", logro organizar, controlar y gestionar ágilmente los datos,

generando los reportes correspondientes con el detalle del pronóstico de carga de energía en el intervalo año-mes-día.

- El uso de herramientas libres PHPAdmyn, MySQL, JAVA, NetBeans, permitieron desarrollar el módulo reduciendo el tiempo y los costes, además de brindar mayor seguridad, fiabilidad y la posibilidad de actualización.
- El uso del módulo de “DISEÑO Y CREACIÓN DEL MÓDULO DE PRONÓSTICO DE CARGA, PARA EL DESARROLLO DE UN SOFTWARE PARA EL ESTUDIO DE REDES DE DISTRIBUCIÓN ELÉCTRICA DE LA CIUDAD DE LOJA”, le permite a la EERSSA la automatización y gestión adecuada de sus actividades.
- El aplicar pruebas de funcionalidad permite determinar si el modulo cumple con los requerimientos planteados por el usuario.
- El desarrollo del módulo permitirá a las futuras promociones de la carrera tanto de sistemas como de electromecánica hacer uso del mismo para generar conocimiento e incertidumbre para mejoras en el mismo a futuro.

i. RECOMENDACIONES

- Utilizar la metodología ICONIX para el desarrollo de software por ser una de las más conocidas y utilizadas en la actualidad, además por ser una herramienta que permite el desarrollo ágil y rápido a través de etapas, que abarcan todo el ciclo de vida de un proyecto.
- La utilización de tecnologías libres como PhpMyAdmyn, el gestor de base de datos MySQL y JAVA NetBeans en el proceso de desarrollo del módulo, porque estas reducen los altos costos de licenciamiento y la poca flexibilidad que presenta el software propietario.
- La utilización del módulo de "DISEÑO Y CREACIÓN DEL MÓDULO DE PRONÓSTICO DE CARGA, PARA EL DESARROLLO DE UN SOFTWARE PARA EL ESTUDIO DE REDES DE DISTRIBUCIÓN ELÉCTRICA DE LA CIUDAD DE LOJA", ya que proporciona a la EERSSA la automatización y gestión de los procesos, permitiéndoles agilizar las actividades que se realizan en la misma.
- Promover el uso del módulo dentro de las carreras que lo necesiten para que se deje de hacer uso de software privativo y que no están relacionados con los datos acercados a la realidad de la EERSSA.
- Brindar talleres de capacitación a las personas que necesiten hacer uso del módulo dentro de la Universidad Nacional de Loja y de la ERRSSA.

j. Bibliografía

[1] L. Bravo, UML y Patrones, Miami: European, 2007.

[2] B. Fernando, El Lenguaje Unificado de Modelado, Londres: Tercera Edición- Adisson Wesley, 2004.

[3] L. Graig, UML y Patrones, España: Segunda Edisión. Lavel S.A., 2003.

[4] M. A. Bustamante Eduardo, Desarrollo de un Sistema para la gestión de versiones de la infraestructura TI, Quito: Escuela Politecnica del Ecuador, 2009.

[5] S. Joseph, Aprendiendo UML en 24 Horas, Florida: Hall Prentice, 1997.

[6] B. Karenny, SELECCIÓN DE METODOLOGÍAS DE DESARROLLO PARA APLICACIONES WEB EN LA FACULTAD DE INFORMÁTICA DE LA UNIVERSIDAD DE CIENFUEGOS, Cuba: Universidad de Cienfuegos “Carlos Rafael Rodríguez”, 2009.

[7] L. R. LEOBARDO, «Metodologia de la programación Orienta a Objetos,» de *Metodologia de la programación Orienta a Objetos*, Barcelona, ALFAOMEGA GRUPO EDITOR, 2010.

[8] V. Martinez, «Repositorios UDA,» 19 julio 2010. [En línea]. Available: http://repositorio.espe.edu.ec/bitstream/21000/4290/1/T-ESPE-032613.pdf. [Último acceso: 27 septiembre 2013].

[9] E. D. D. C. N. D. ELECTRICIDAD, REGULACION No. CONELEC – 004/01, Quito, 23 de mayo de 2001.

[10] R. Carvajal, Redistribución de transformadores, Habana Cuba: Cujae, 2010.

[11] T. Gómez, La Regulación del Sector Eléctrico, Nuevas Tendencias, Cartagena: Proyecto CREG, 2009.

[12] R. Samuel, Redes de Distribución de Energía, Colombia: Tercera Ediación, Universidad Nacional de Colombia sede Manizales, 2004.

[13] E. E. d. M. C.A., Diseño del Sistema de Distribución Primario de la Subestación Sur, Milagro, 2000.

[14] R. Peñafiel, Diseño y Construcción de Redes Eléctricas, Cuenca: Tesis de Ingeniería Eléctrica, 2013.

[15] O. P. Mondragón, Pronóstico de la Demanda de Energía Eléctrica A Corto Plazo Utilizando Una Red Neuronal, Mexico: Instituto Politecnico Nacional, 2011.

[16] S. Rosales, «Instituto Universitario de Tecnología "Dr. Federico Rivero Palacio",» 07 febrero 2011. [En línea]. Available: http://bibdigital.iut.frp.ven/bitstream/15000/1481/1/CD-2213.pdf. [Último acceso: 24 septiembre 2013].