

Universidad de las Ciencias Informáticas

Facultad 10



Título: “Guía para la obtención de la adaptabilidad, instalabilidad y coexistencia de las soluciones software de apoyo a la toma de decisiones basadas en Data Warehouse”

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

Autor(es): Liyelis Betancourt Molina

Eilen Sánchez Padilla

Tutor: Ing. Nadia Porro Lugo

Co-tutor: Msc. Michael González Jarrín

Ciudad de La Habana, Cuba, 2010.

“Año del 51 Aniversario del Triunfo de la Revolución”



“No sobrevive el más fuerte de la especie, ni el más inteligente, sino el más adaptable a los cambios”

Charles Darwin

Declaración de autoría

Declaramos que somos las únicas autoras de este trabajo y autorizamos a la Facultad 10 de la Universidad de las Ciencias Informáticas; así como a dicho centro a consultar cualquier cambio que se decida aplicar al mismo con su autor.

Para que así conste firmamos la presente a los _____ días del mes de _____ del año 2010.

Eilen Sánchez Padilla

Liyelis Betancourt Molina

Ing. Nadia Porro Lugo

Agradecimientos

Agradecemos a todas aquellas personas que hicieron posible la realización de este trabajo.

Agradecimientos de Eilen:

A mis amigos que nunca han dejado de animarme y aconsejarme.

A mis compañeros de aula, por brindarme su apoyo cuando más lo necesitaba en la carrera.

A mi querido y amado Armando (Tito) por estar siempre a mi lado en los momentos más difíciles de mi vida.

Agradecimientos de Liyelis:

A mis padres Vivian y Lázaro, por ser mi bastón y siempre darme su apoyo en cada evento de mi vida, por la confianza que existe entre nosotros, son mi gran tesoro.

A Yadier por ser la persona más importante en mi vida, por no darme la espalda nunca, por ser mi amigo, mi amor, mi compañero.

A mis compañeras de l apartamento 121 206, a Eilen, a Dalvis, a Elibet, a Anayvis, a Dayana, a Yude y a Yelena, por que son las mujeres más divertidas y sinceras que he conocido.

A mi amiga Loida, a Lianet, a Yanlay, al Chino, a Surama, por que siempre me dicen que si, los quiero mucho.

Dedicatoria

Eilen Sánchez Padilla:

Quisiera dedicar este trabajo a toda mi familia, a mis hermanos, los quiero mucho, son la base de mi vida para salir adelante y triunfar, también le dedico la tesis a mi querido Tito.

Liyelis Betancourt Molina:

Quisiera dedicar este trabajo a toda mi familia, especialmente a Lázaro y a Vivian, que son mis padres, por estar ahí siempre conmigo, por sufrir cuando lloro, por reír cuando estoy feliz.

Quisiera dedicar este trabajo a una persona especial, a mi inolvidable Yadier, por ser siempre esa mano amiga que me apoya en todo, que me impulsa hacia el camino correcto, jamás te olvidaré.

Índice General

Índice de figuras	VIII
Índice de tablas.....	IX
Resumen	X
Introducción	1
1. CAPÍTULO 1: Fundamentación Teórica.....	5
1.1 Introducción.....	5
1.2 Conceptos asociados al problema.....	5
1.3 Guía.....	5
1.4 Calidad.....	6
1.4.1 Concepto de Calidad	6
1.4.2 Calidad del Software.....	6
1.4.3 Modelos que determinan la Calidad del Software	7
1.5 Base de datos	13
1.5.1 Definición de las Bases de Datos.....	13
1.5.2 Tipos de Base de Datos	13
1.5.3 Modelos para el diseño de las Bases de Datos	15
1.6 Sistema Gestor de Base de Datos.....	20
1.6.1 Algunos Gestores de Bases de Datos	20
1.7 Sistemas de soporte a la toma de decisiones (DSS), buenas prácticas, proceso de desarrollo de software.	24
1.7.1 Qué es un sistema de Soporte a la Toma de Decisiones.	24
1.7.2 Clasificación de los DSS.	24
1.7.3 ¿Qué es una buena práctica?.....	28
1.7.4 Proceso de desarrollo de software.....	28
1.6 Conclusiones.....	28
2. Capítulo 2: Propuesta de solución	29
2.1 Algunas características de la Línea de Soluciones de Almacenes de Datos e Inteligencia de Negocio en DATEC	29

2.2 Factores que afectan la portabilidad.....	30
2.4 Propuesta de la guía de solución.....	34
2.5 Conclusiones.....	49
3. Capítulo 3: Validación de la propuesta.....	50
3.1 Métodos de Validación	50
3.1.2 Comparación de los Resultados de Salida del Modelo con los del Sistema Real:	50
3.2 Elaboración del Cuestionario para la Validación de la Propuesta	54
3.2.1 Resultados de la evaluación a través del método Delphi.....	54
3.3 Conclusiones.....	62
Conclusiones generales.....	63
Recomendaciones	64
Referencias bibliográficas.....	65
Bibliografía.....	67
Anexos.....	70
Glosario de términos.....	74

Índice de figuras

Figura 1: Modelo de calidad para la calidad externa e interna.....	10
Figura 2: Ejemplo de estructura del Modelo Jerárquico.....	15
Figura 3: Ejemplo de estructura del Modelo de Red	16
Figura 4: Contenido de un DSS.....	27
Figura 5: Ciclo de vida de las soluciones software.....	35
Figura 6: Ubicación de los factores.....	55
Figura 7: Efectividad de la guía.....	56
Figura 8: Necesidad de una guía.....	56
Figura 9: Nivel de complejidad de una guía.....	57
Figura 10: Nivel de satisfacción de la guía.....	58
Figura 11: Repercusión de la guía.....	59
Figura 12: Adaptabilidad de la guía.....	59
Figura 13: Nivel de cumplimiento.....	60
Figura 14: Por ciento de aceptación.....	61
Figura 15: Coeficiente de Kendall.....	62

Índice de tablas

Tabla 1: Herramientas, lenguajes y metodología.....	29
Tabla 2: Los factores que afectan las subcaracterísticas de portabilidad.....	34
Tabla 3: Tabla de factores ubicados en las fases de la guía.....	48

Resumen

El presente trabajo se desarrolló en la facultad 10 de la Universidad de las Ciencias Informáticas. Tiene como objetivo la creación de una guía para la obtención de la adaptabilidad, instalabilidad y coexistencia de las soluciones software de apoyo a la toma de decisiones basadas en Data Warehouse del Centro Tecnologías de Gestión de Datos (DATEC).

Para la definición de esta guía se realizó un estudio del estado del arte acerca de la adaptabilidad, instalabilidad y coexistencia de las soluciones software, con el objetivo de lograr un mejor entendimiento del trabajo en general, así como se analizan conceptos emitidos por diferentes autores, para obtener una propuesta final válida y eficaz. También se analizan algunas de las definiciones como: Calidad, Calidad del Software, Sistema Gestores de Base de Datos, entre otras. Se realizó el método de la entrevista para determinar los factores de las soluciones software que se crean en DATEC.

Luego de un intenso análisis realizado se declaró la propuesta de solución para el cumplimiento del objetivo trazado, dicha propuesta fue guiada por el modelo ISO-9126, donde está orientada por el factor de calidad portabilidad, así como por sus principales subcaracterísticas: adaptabilidad, instalabilidad y coexistencia. Luego se realizó el Método de Experto a través de la encuesta para validar dicha propuesta.

Con esta guía los proyectos tendrán una forma de elevar y mejorar la calidad de sus soluciones software, así como obtener la portabilidad para adaptarlos, instalarlos, y que sean capaces de coexistir con otros recursos.

Palabras clave: Calidad, SGBD, Adaptabilidad, Bases de Datos Data Warehouse, Instalabilidad, PostgreSQL, Portabilidad, Coexistencia, Calidad de Software.

Introducción

Los software optimizados para la administración de base de datos constituyen la interfaz entre la computadora y el hombre, lo mismo que entre las personas. Por una parte, hacen posible las operaciones de las empresas modernas, pues posibilitan la manera más fácil de organizar los datos y lograr el control de las operaciones, así como la toma de decisiones y la mejora, tanto de la eficacia de los procedimientos como de la atención al cliente.

En el mundo de los procesos, se cree en la eficiencia de controlarlos y llevar un seguimiento para poder optimizarlos, pero la época en la que estamos viviendo no se rige por estos fundamentos, ya que el desarrollo de aplicaciones se ha convertido de cierta forma imprescindible porque ha resultado imposible controlar el cambio constante de las variables del entorno, por lo cual el desarrollo del software no puede verse desde una perspectiva lineal en ningún caso.

En Cuba la creación de una guía para la elaboración de software que sean portables sería de gran utilidad, ya que en la mayoría de las empresas y centros de trabajo se haría más fácil la manipulación y el control tanto de la información propia de la entidad como de los datos correspondientes al personal de trabajo de las mismas.

La Universidad de las Ciencias Informáticas, no está exenta de dicha necesidad, pues en DATEC se ha investigado que las soluciones software presentan conflictos en cuanto a la portabilidad, por lo tanto se desconoce qué capacidad de adaptación adquiere el software en un entorno e incluso el esfuerzo que requiere el usuario para poder adaptarlo. Además, no se percibe cual sería el comportamiento del software al coexistir con otros en un sistema específico y por tanto el esfuerzo que se requiere para instalarlo. Todo lo planteado trae como consecuencia que existan dificultades o incongruencias en la interacción con el cliente de las soluciones software, en el proceso de las pruebas de aceptación, lo que afecta en gran medida la credibilidad como productores software.

De la anterior situación problémica se determina que el **problema científico** a resolver es el siguiente:

¿Cómo obtener la adaptabilidad, la instalabilidad y la coexistencia de las soluciones de software de apoyo a la toma de decisiones basadas en las bases de datos Data Warehouse del Centro Tecnologías de Gestión de Datos (DATEC)?

El **objeto de estudio** de este trabajo, estará centrado en el estudio de la adaptabilidad, la instalabilidad y la coexistencia, teniendo como **campo de acción** la adaptabilidad, la instalabilidad y la coexistencia de las soluciones de software de apoyo a la toma de decisiones basadas en Data Warehouse del Centro Tecnologías de Gestión de Datos.

El **objetivo general** es elaborar una guía para obtener la adaptabilidad, instalabilidad y coexistencia de las soluciones de software de apoyo a la toma de decisiones basadas en Data Warehouse del centro DATEC.

Objetivos específicos:

- Elaborar el marco teórico de la investigación.
- Elaborar una guía para obtener la adaptabilidad, la instalabilidad y la coexistencia de las soluciones de software de apoyo a la toma de decisiones basadas en Data Warehouse del centro DATEC.
- Validar la propuesta.

Idea a defender:

Con la elaboración de una guía se obtendrá la adaptabilidad, instalabilidad y la coexistencia en las soluciones software de apoyo a la toma de decisiones basadas en Data Warehouse del centro DATEC.

Tareas a realizar:

- Seleccionar y revisar la bibliografía para actualizar los logros y limitaciones existentes que se obtengan de la adaptabilidad, instalabilidad y coexistencia de las soluciones software de apoyo a la toma de decisiones basadas Data Warehouse del Centro Tecnologías de Gestión de Datos que se basan en PostgreSQL.
- Analizar los modelos de calidad tales como ISO/IEC 9126, McCall, Bohem, así como analizar el gestor de base de datos PostgreSQL.
- Identificar los involucrados potenciales del Centro Tecnologías de Gestión de Datos para la realización de un diagnóstico que permita conocer el estado actual del tema.
- Analizar los factores asociados con las subcaracterísticas de calidad estudiadas para la determinación de los problemas más frecuentes.
- Determinar unas buenas prácticas para obtener la adaptabilidad, instalabilidad y coexistencia de las soluciones de software.

- Aplicar la guía a soluciones de software, para determinar si ha cambiado el estado inicial al obtener adaptabilidad, instalabilidad y coexistencia de las soluciones software.

Métodos científicos:

El método científico de investigación es la forma de estudiar los fenómenos de la naturaleza y la sociedad para descubrir sus relaciones y su esencia, el mismo se puede clasificar en métodos teóricos y métodos empíricos los cuales están relacionados entre sí de forma dialéctica. [20]

En el documento se muestran los métodos que se utilizan para el desarrollo de la investigación.

Métodos Teóricos:

Existen características que no son observables directamente dentro del problema y se pueden estudiar a través de los siguientes métodos:

Método Histórico-Lógico: Apoyándose en el estudio del problema, se manifiesta la esencia del mismo, explicando su historia y se ha llevado a cabo un estudio del origen y desarrollo de la adaptabilidad, instalabilidad y coexistencia de las soluciones software de apoyo a la toma de decisiones basadas en Data Warehouse en el Centro Tecnologías de Gestión de Datos basadas en PostgreSQL.

Análisis y Síntesis: Para un mejor entendimiento y estudio del problema se ha descompuesto el mismo en varios elementos, de esta forma se ha concretado cuáles son los fenómenos que lo generan. Luego de haber realizado el análisis de todos los fragmentos del problema se hace una unión del mismo, posibilitando definir sus características generales y las relaciones que existen entre las partes analizadas. [19]

Métodos Empíricos:

Encuesta: Se realiza cuando la información que se requiere puede ser obtenida a partir de la respuesta que una o varias personas puedan dar a un cuestionario pre elaborado, y las mismas están dispuestas a colaborar con la investigación. En el presente trabajo se realizó una encuesta con el objetivo de validar las principales actividades a tener presente en la elaboración de la guía para obtener la adaptabilidad, instalabilidad y coexistencia de las soluciones software de apoyo a la toma de decisiones basadas en Data Warehouse. [21]

Entrevista: Se efectúa para conocer el fenómeno y tiene como objetivo obtener toda la información necesaria del problema. Entonces se realizaron las entrevistas

necesarias a los usuarios que interactúan con las soluciones software en la línea de Soluciones de almacenes de datos e inteligencia de negocio de DATEC, lo cual permitió tener un mejor entendimiento de la situación y de esa forma se puede determinar una propuesta de solución a la problemática existente. [19]

Se espera que con la realización de este trabajo las soluciones software desarrolladas en los proyectos productivos de la universidad logren alcanzar gran adaptabilidad, instalabilidad y coexistencia.

El actual trabajo de diploma estará organizado en tres capítulos:

Capítulo 1, Fundamentación Teórica: En este capítulo se abordan los principales conceptos a tener en cuenta durante todo el desarrollo del trabajo (Calidad, Calidad del Software, Estándares, Modelos de Calidad, Sistemas Gestores de Base de Datos, entre otros), además se realiza un estudio del estado del arte de las soluciones de software de apoyo a la toma de decisiones basadas en Data Warehouse del centro DATEC, por lo tanto se analizan las características del Gestor de Base de Datos PostgreSQL, porque es el Gestor de Base de Datos que se utiliza en dicho centro.

Capítulo 2, Propuesta de Solución: En este capítulo se define la guía para la obtención de la adaptabilidad, instalabilidad y coexistencia de las soluciones de software de apoyo a la toma de decisiones basadas en Data Warehouse del centro DATEC.

Capítulo 3, Validación de la Solución Propuesta: Capítulo donde se realizará la validación de la propuesta con el método Delphi basado en el criterio de experto.

1. CAPÍTULO 1: Fundamentación Teórica

1.1 Introducción

La gran variedad de software que se producen o que se han producido debe cumplir múltiples requisitos, porque cuando se desarrolla un software es importante saber manejar los problemas comunes que puedan presentarse, como por ejemplo, el cambio de los requerimientos o de ámbito de las aplicaciones, así como la introducción de nuevas tecnologías, mientras se desarrolla una aplicación, lo cual es muy normal, debido a lo volátil que son las organizaciones hoy en día. Los grandes programas deben poder aplicarse a una amplia gama de condiciones y requisitos, adaptarse a entornos laborales y financieros cambiantes y tener la capacidad de ajustarse a las necesidades de empresas pequeñas y grandes, además de conservar su utilidad durante un período de tiempo prolongado, pues se espera que tengan una larga vida útil y sean muy seguros, por tanto la adaptabilidad, instalabilidad y coexistencia del software son vitales como infraestructura de la sociedad moderna. En el presente capítulo se explican conceptos relacionados principalmente con la Calidad y las Normas y, Estándares del Software con el objetivo de lograr que las soluciones software satisfagan de forma más eficiente a los clientes, estos términos, conjunto con otros aquí abordados servirán de soporte teórico a la investigación desarrollada y contribuirán a un mejor entendimiento del trabajo en general.

1.2 Conceptos asociados al problema

El lector debe llegar a entender el problema que se aborda en el trabajo y para que este obtenga una buena comprensión del mismo, se le proporcionará el trayecto más adecuado posible, detallando conceptos asociados al problema, entre los que se encuentran: guía, calidad, base de datos, Data Warehouse, sistema gestor de base de datos, PostgreSQL, toma de decisiones, buenas prácticas, proceso de desarrollo del software, entre otros.

1.3 Guía

Conjunto de criterios bien definidos y documentados que encaminan una actividad o tarea. [23]

Ayuda a desarrollar o implementar un plan de garantía de calidad para un proyecto. Esta tarea es propia del personal de Garantía de Calidad (GC). Para iniciar la puesta en práctica de procedimientos de GC en un proyecto, esta tarea es propia de los Jefes de Proyecto de Desarrollo de Software y para evaluar o especificar un plan de garantía de calidad, es propia de compradores o usuarios. [23]

1.4 Calidad

En la sección anterior se definió que es una guía, donde se menciona la GC, por lo tanto considerando la amplia aplicación y el uso constante en la actualidad de las soluciones software, se necesita que los mismos cuenten con toda la calidad que los usuarios exigen, entonces vale señalar la necesidad de introducir algunos conceptos de este tema tan significativo.

1.4.1 Concepto de Calidad

La norma ISO 8402 define Calidad como el conjunto de características de un elemento que le confieren la aptitud para satisfacer necesidades explícitas e implícitas. [12]

La Sociedad Americana para el Control de Calidad (ASQC, American Society for Quality Control), la define como el conjunto de características de un producto, proceso o servicio que le confieren su aptitud para satisfacer las necesidades del usuario o cliente. [5]

Edward Deming consultor y difusor del concepto de calidad y a quien se le acredita parte del éxito de las industrias japonesas, en 1986 define la Calidad como: “predecible grado de uniformidad, a bajo costo y útil para el mercado”.

1.4.2 Calidad del Software

Después de haber estudiado algunas definiciones de Calidad se analiza la importancia de la calidad del software. En los estudios realizados a través de los años varios autores la han catalogado de diferentes maneras:

El profesor Roger S. Pressman uno de los grandes investigadores de la Ingeniería de Software define la Calidad del Software como: “la concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente”. [13] También se plantea que la Calidad del Software es el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o

usuario. Conjunto de cualidades que lo caracterizan y que determinan su utilidad y existencia. Conjunto de propiedades y de características de un producto o servicio, que le confieren aptitud para satisfacer una necesidad explícita o implícita. [14]

1.4.3 Modelos que determinan la Calidad del Software

Una vez conocido las definiciones de calidad, y calidad del software se puede pasar al estudio de los modelos que determinan la calidad del software.

Un modelo de calidad no es más que las técnicas, herramientas y metodologías que le facilitan a las empresas que se encargan de la fabricación de software, guiar por un camino único el avance de dicho proceso y así lograr que se cumpla con los requisitos iniciales pedidos por el cliente ya que esa es la base de la calidad de un producto. [25]

Existen diferentes modelos que determinan la calidad del software, entre estos modelos tenemos:

- McCall.
- Bohem.
- ISO-9126.

Según McCall

Este modelo evalúa el software desde tres puntos de vista distintos: Operación del producto (utilizándolo), revisión del producto (cambiándolo) y transición del producto (portándolo). Los mismos se clasifican según sus características en: [16]

1. Operaciones del producto: Características operativas

- Corrección (¿Hace lo que se le pide?)
- Fiabilidad (¿Lo hace de forma fiable todo el tiempo?)
- Eficiencia (¿Qué recursos hardware y software necesito?)
- Integridad (¿Puedo controlar su uso?)
- Facilidad de uso (¿Es fácil y cómodo de manejar?)

2. Revisión del producto: Capacidad para soportar cambios

- Facilidad de mantenimiento (¿Puedo localizar los fallos?)
- Flexibilidad (¿Puedo añadir nuevas opciones?)
- Facilidad de prueba (¿Puedo probar todas las opciones?)

3. Transición del producto: Adaptabilidad a nuevos entornos

- Portabilidad (¿Podré usarlo en otra máquina?)
- Reusabilidad (¿Podré utilizar alguna parte del software en otra aplicación?)
- Interoperabilidad (¿Podrá comunicarse con otras aplicaciones o sistemas informáticos?)

Según Bohem

El segundo modelo de calidad más conocido es el presentado por Barry Bohem en 1978, este modelo introduce características de alto nivel, características de nivel intermedio y características primitivas, cada una de las cuales contribuye al nivel general de calidad.

Las características de alto nivel representan requerimientos generales de uso, que son:

- utilidad (usable, confiable, eficiente) es el producto en sí mismo.
- mantenibilidad cuán fácil es modificarlo, entenderlos y retestearlo.
- utilidad general si puede seguir usándose si se cambia el ambiente.

Las características del nivel intermedio representan los factores de calidad de Bohem:

- portabilidad (utilidad general).
- confiabilidad (utilidad).
- eficiencia (utilidad).
- usabilidad (utilidad).
- testeabilidad (mantenibilidad).

- facilidad de entendimiento (mantenibilidad).
- modificabilidad o flexibilidad (mantenibilidad).

El nivel más bajo que son las características primitivas corresponde a características directamente asociadas a uno o dos métricas de calidad:

- de portabilidad: independencia de dispositivo, auto-contención.
- de confiabilidad: auto-contención, exactitud, completitud, consistencia, robustez/integridad.
- de eficiencia: accesibilidad, eficiencia de uso de dispositivo.
- de usabilidad: robustez/integridad, accesibilidad, comunicación.
- de testeabilidad: comunicación, auto descripción, estructuración.
- de entendibilidad: consistencia, estructuración, concisión, legibilidad.
- de modificabilidad: estructuración, aumentabilidad.

Según ISO – 9126

En 1992, una variante del modelo de McCall fue propuesta como estándar internacional para medición de calidad de software, ISO- 9126 (Software Product Evaluation: Quality Characteristics and Guidelines for their Use) , el cual establece que cualquier componente de la Calidad del Software puede ser descrito en términos de una o más de seis características básicas, las cuales son: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad; cada una de las cuales se detalla a través de un conjunto de sub-características que permiten profundizar en la evaluación de la Calidad de productos de software.[11]

A continuación se muestra una imagen que representa las características y subcaracterísticas en que se divide el modelo:

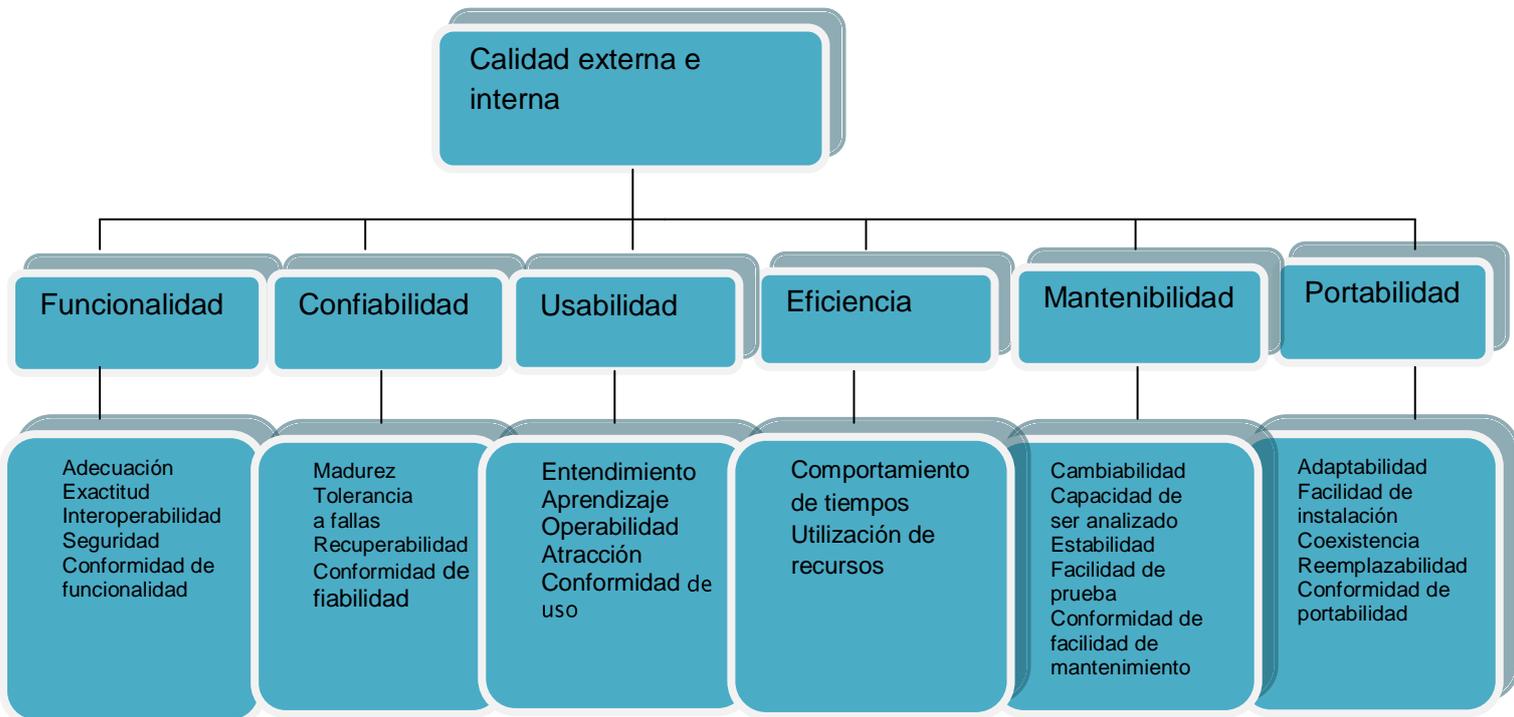


Figura 1: Modelo de calidad para la calidad externa e interna

Luego se detalla cada una de las características que establece este modelo:

1. **Funcionalidad:** En este grupo se conjunta una serie de atributos que permiten calificar si un producto de software maneja en forma adecuada el conjunto de funcionalidades que satisfagan las necesidades para las cuales fue diseñado. Para este propósito se establecen los siguientes atributos:

- **Adecuación.** Se enfoca a evaluar si el software cuenta con un conjunto de funciones apropiadas para efectuar las tareas que fueron especificadas en su definición.
- **Exactitud.** Este atributo permite evaluar si el software presenta resultados o efectos acordes a las necesidades para las cuales fue creado.
- **Interoperabilidad.** Permite evaluar la habilidad del software de interactuar con otros sistemas previamente especificados.
- **Conformidad.** Evalúa si el software se adhiere a estándares, convenciones o regulaciones en leyes y prescripciones similares.

- Seguridad. Se refiere a la habilidad de prevenir el acceso no autorizado, ya sea accidental o premeditado, a los programas y datos.

2. Confiabilidad: Aquí se agrupan un conjunto de atributos que se refieren a la capacidad del software de mantener su nivel de ejecución bajo condiciones normales en un periodo de tiempo establecido. Las sub-características que el estándar sugiere son:

- Nivel de Madurez. Permite medir la frecuencia de falla por errores en el software.
- Tolerancia a fallas. Se refiere a la habilidad de mantener un nivel específico de funcionamiento en caso de fallas del software o de cometer infracciones de su interfaz específica.
- Recuperación. Se refiere a la capacidad de restablecer el nivel de operación y recobrar los datos que hayan sido afectados directamente por una falla, así como el tiempo y el esfuerzo necesarios para lograrlo.

3. Usabilidad: Consiste en un conjunto de atributos que permiten evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar el sistema.

- Comprensibilidad. Se refiere al esfuerzo requerido por los usuarios para reconocer la estructura lógica del sistema y los conceptos relativos a la aplicación del software.
- Facilidad de Aprender. Establece atributos del software relativos al esfuerzo que los usuarios deben hacer para aprender a usar la aplicación.
- Operabilidad. Agrupa los conceptos que evalúan la operación y el control del sistema.

4. Eficiencia: Esta característica permite evaluar la relación entre el nivel de funcionamiento del software y la cantidad de recursos usados. Los aspectos a evaluar son:

- Comportamiento con respecto al tiempo. Atributos del software relativos a los tiempos de respuesta y de procesamiento de los datos.

- Comportamiento con respecto a recursos. Atributos del software relativos a la cantidad de recursos usados y a la duración de su uso en la realización de sus funciones.

5. **Mantenibilidad:** Se refiere a los atributos que permiten medir el esfuerzo necesario para realizar modificaciones al software, ya sea por la corrección de errores o por el incremento de funcionalidad. En este caso, se tienen los siguientes factores:

- Capacidad de análisis. Relativo al esfuerzo necesario para diagnosticar las deficiencias o causas de fallas, o para identificar las partes que deberán ser modificadas.
- Capacidad de modificación. Mide el esfuerzo necesario para modificar aspectos del software, remover fallas o adaptar el software para que funcione en un ambiente diferente.
- Estabilidad. Permite evaluar los riesgos de efectos inesperados debidos a las modificaciones realizadas al software.
- Facilidad de Prueba. Se refiere al esfuerzo necesario para validar el software una vez que fue modificado.

6. **Portabilidad:** En este caso, se refiere a la habilidad del software de ser transferido de un ambiente a otro, y considera los siguientes aspectos:

- Adaptabilidad. Evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones.
- Facilidad de Instalación. Es el esfuerzo necesario para instalar el software en un ambiente determinado.
- Conformidad. Permite evaluar si el software se adhiere a estándares o convenciones relativas a portabilidad.
- Capacidad de reemplazo. Se refiere a la oportunidad y el esfuerzo usado en sustituir el software por otro producto con funciones similares.
- Coexistencia. Es la capacidad del producto del software para coexistir con otros productos de software independientes dentro de un mismo entorno, compartiendo recursos comunes.

1.5 Base de datos

A continuación se muestra un estudio sobre las Bases de Datos (BD), luego de haber analizado las definiciones de guía, calidad, así como la calidad del software y los diferentes modelos que determinan la misma. El análisis va a estar centrado principalmente en las base de datos Data Warehouse, debido a que son la base para el desarrollado de software de apoyo a la toma de decisiones.

1.5.1 Definición de las Bases de Datos

Se entiende por base de datos:

Una BD es un conjunto de datos interrelacionados entre sí, almacenados con carácter más o menos permanente en la computadora, o sea, una BD puede considerarse una colección de datos variables en el tiempo, un sistema de archivos electrónico.[9]

Una BD es un conjunto de datos almacenados entre los que existen relaciones lógicas y ha sido diseñada para satisfacer los requerimientos de información de una empresa u organización. En una BD, además de los datos, también se almacena su descripción. [4]

1.5.2 Tipos de Base de Datos

Las BD pueden clasificarse de varias maneras, de acuerdo al criterio a elegir.

Según la variabilidad de los datos almacenados pueden clasificarse en:

- **Bases de datos estáticas:** Son BD de sólo lectura, utilizadas primordialmente para almacenar datos históricos que posteriormente se pueden utilizar para estudiar el comportamiento de un conjunto de datos a través del tiempo, realizar proyecciones y tomar decisiones. Son aquellas BD en las que no se puede modificar la información que está guardada, es decir, solo se puede consultar.[29]

- **Bases de datos dinámicas:** Son BD donde la información almacenada se modifica con el tiempo, permitiendo operaciones como actualización y adición de datos, además de las operaciones fundamentales de consulta. Son aquellas BD en las que la información que está almacenada está en constante cambio, ya que sobre ella se pueden realizar operaciones de inserción, actualización y eliminación así como las operaciones de consultas. [29]

Según el contenido pueden clasificarse en:

- **Bases de datos bibliográficas:** Es un conjunto de referencias bibliográficas de publicaciones almacenadas informáticamente y que pueden ser recuperadas interactivamente gracias al lenguaje de consultas.
- **Bases de datos de texto completo:** Contienen el texto completo de la fuente de documentos que comprende la BD. Almacenan las fuentes primarias, por ejemplo, todo el contenido de las ediciones de una colección de revistas científicas.
- **Directorios:** Proporcionan información sobre individuos, organizaciones y servicios, ejemplos nombres, direcciones entre otras cosas. Un ejemplo son las guías telefónicas en formato electrónico. [8]
- **Bases de datos o "bibliotecas" de información Biológica:** Son bases de datos que almacenan diferentes tipos de información proveniente de las ciencias de la vida o médicas. Se pueden considerar en varios subtipos:
 1. Aquellas que almacenan secuencias de nucleótidos o proteínas.
 2. Las bases de datos de rutas metabólicas.
 3. Bases de datos de estructura, comprende los registros de datos.
 4. Experimentales sobre estructuras 3D de biomoléculas.
 5. Bases de datos clínicas.
 6. Bases de datos bibliográficas (biológicas). [29]

1.5.3 Modelos para el diseño de las Bases de Datos

Además de la clasificación por la función de las bases de datos, éstas también se clasifican de acuerdo a su modelo de administración de datos. Las BD a través de su evolución han atravesado diferentes estructuras, desde los modelos de red hasta los modelos actuales, cada uno de estos modelos tiene características peculiares que los identifican además de la fecha en que surgen, estos modelos se pueden agrupar en dos grandes grupos: los modelos tradicionales (Modelo jerárquico, modelo de red, modelo relacional) y los modelos avanzados (de los modelos orientados a objetos, Data Warehouse). [30]

Modelo de Bases de Datos Jerárquicas

Son bases de datos que, como su nombre indica, almacenan su información en una estructura jerárquica. Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento. [29]

El modelo jerárquico fue desarrollado para permitir la representación de aquellas situaciones de la vida real en las que predominan las relaciones de tipo 1: N. Es un modelo muy rígido en el que las diferentes entidades de las que está compuesta una determinada situación, se organizan en niveles múltiples de acuerdo a una estricta relación PADRE/HIJO, de manera que un padre puede tener más de un hijo, todos ellos localizados en el mismo nivel, y un hijo únicamente puede tener un padre situado en el nivel inmediatamente superior al suyo. [15] Esta estricta relación PADRE/HIJO implica que no puedan establecerse relaciones entre segmentos dentro de un mismo nivel siendo esta una de las limitaciones, además de representar eficientemente la redundancia de datos. Véase en la siguiente figura: [8]

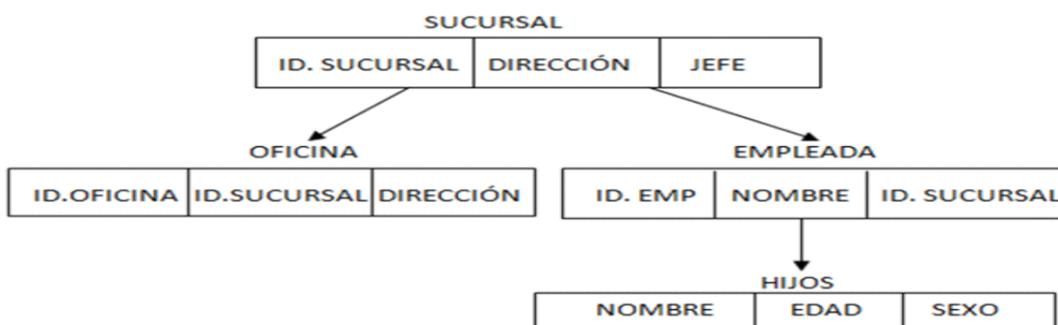


Figura 2: Ejemplo de estructura del Modelo Jerárquico.

Modelo de Bases de Datos de Red

El modelo de datos de red aparece a mediados de los 60 como respuesta a limitaciones del modelo jerárquico en cuanto a representación de relaciones más complejas. Es un modelo ligeramente distinto del jerárquico; su diferencia fundamental es la modificación del concepto de nodo ya que el modelo de red permite que un nodo tenga varios nodos padres, por lo tanto representa las entidades en forma de nodos de un grafo, y las interrelaciones entre estas mediante arcos que unen dichos nodos. En principio esta representación no impone restricción alguna acerca del tipo y el número de arcos que puede haber, con lo que se pueden modelar estructuras de datos tan complejas como sea necesario. [6]

Esto ofrece una solución eficiente al problema de la redundancia de datos. Es un modelo poco utilizado actualmente. En la siguiente figura se puede apreciar la estructura de una BD de red:

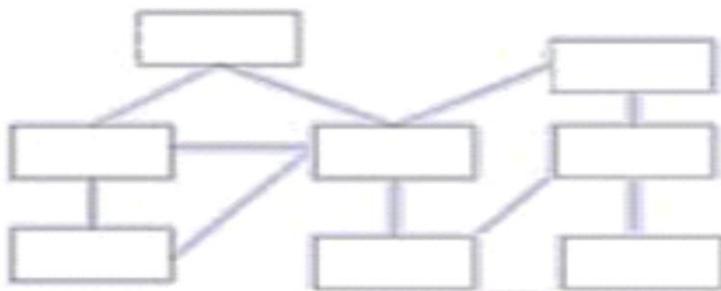


Figura 3: Ejemplo de estructura del Modelo de Red.

Modelo de Base de Datos Declarativo

Este modelo usado para bases de conocimiento, que no son más que bases de datos con mecanismos de consulta en los que el trabajo de extracción de información a partir de los datos recae en realidad sobre el ordenador, en lugar de sobre el usuario. Estos mecanismos de consulta exigen que la información se halle distribuida de manera que haga eficiente las búsquedas de los datos, ya que normalmente las consultas de este tipo requieren acceder una y otra vez a los datos en busca de patrones que se adecuen a las características de los datos que ha solicitado el usuario. Entre las bases

de datos declarativas se puede citar fundamentalmente dos: las deductivas, y las funcionales. Ambas extienden paradigmas o métodos de programación (al igual que ocurre con la programación orientada a objetos) a las bases de datos, de manera que ambos, programa y base de datos puedan cooperar más eficientemente en la resolución del problema. [30]

Modelo Bases de Datos Documentales

Permiten la indexación a texto completo, y en líneas generales realizar búsquedas más potentes. [30]

Modelo de Bases de Datos Distribuida

La base de datos está almacenada en varias computadoras conectadas en red. Surgen debido a la existencia física de organismos descentralizados. Esto les da la capacidad de unir las bases de datos de cada localidad y acceder así a distintas universidades, sucursales de tiendas. [30]

Modelo de Bases de Datos Orientadas a Objetos

Este modelo es bastante reciente y propio de los modelos informáticos orientados a objetos, trata de almacenar en la BD los objetos completos (estado y comportamiento), las cuales se les conoce como Base de Datos Orientadas a Objetos (BDOO), donde los elementos de datos son objetos y las relaciones se mantienen por medio de inclusión lógica. Las entidades están representadas como clases, además contiene un método sistemático de representación de relación, y la interfaz uniforme de usuario es un sistema de mensajes que puede explorar los objetos y sus interconexiones. [29]

Modelo de Base de Datos Relacional

Éste es el modelo más utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Tras ser postulados sus fundamentos en 1970 por Edgar Frank Codd, de los laboratorios IBM en San José (California), no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos. Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados tuplas. Se piensa en cada relación como si fuese una tabla que está compuesta por registros (las filas de una tabla), que

representarían las tuplas, y campos (las columnas de una tabla). En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. Cada relación tiene un o un conjunto de atributos, conocidos como clave primaria que identifican unívocamente a la fila. Las referencias de interrelaciones son manipuladas a través de llaves foráneas, que son punteros lógicos de una fila de una relación a una fila de otra relación. Para estar seguro de la validación de los datos en una base de datos relacional, Codd formuló las reglas de integridad de la entidad y las reglas de integridad referencial. Estas reglas establecen que el valor de una clave primaria no puede ser nulo, y que el valor de una clave externa debe corresponder con el valor actual de la clave primaria de la otra relación a la que se hace referencia. [7]

Data Warehouse

Para facilitar la comprensión del término Data Warehouse (DW), se hace necesario hacer referencia a diferentes definiciones. La más conocida fue propuesta por Inmon [17] (considerado el padre de las Bases de Datos) en 1992: “Un DW es una colección de datos orientados a temas, integrados, no-volátiles y variante en el tiempo, organizados para soportar necesidades empresariales”. En 1993, Susan Osterfeldt [17] publica una definición que sin duda acierta en la clave del DW: “Yo considero al DW como algo que provee dos beneficios empresariales reales: Integración y Acceso de datos. DW elimina una gran cantidad de datos inútiles y no deseados, como también el procesamiento desde el ambiente operacional clásico”. Esta última definición refleja claramente el principal beneficio que el Data Warehouse aporta a la empresa, eliminar aquellos datos que obstaculizan la labor de análisis de información y entregar la información que se requiere en la forma más apropiada, facilitando así el proceso de gestión.

Los Data Warehouse surgen con la promesa del manejo y control de la información. Ellos aseguran una vista única de los datos, que pueden provenir de diversas fuentes. Gracias a esto, los usuarios finales no se ven en la necesidad de aprender y utilizar múltiples sistemas de acceso y manipulación de los datos. Un Data Warehouse facilita la comprensión de los datos, transformándolos en información útil, teniendo como bandera el apoyo a la Toma de Decisiones. [17] Ver anexo 1.

Dentro de las facilidades de los Data Warehousing se encuentran que:

- Proporciona una herramienta para la toma de decisiones en cualquier área funcional, basándose en información integrada y global del negocio.
- Facilita la aplicación de técnicas estadísticas de análisis y modelización para encontrar relaciones ocultas entre los datos del almacén obteniendo un valor añadido para el negocio de dicha información.
- Proporciona la capacidad de aprender de los datos del pasado y de predecir situaciones futuras en diversos escenarios.
- Simplifica dentro de la empresa la implantación de sistemas de gestión integral de la relación con el cliente.
- Supone una optimización tecnológica y económica en entornos de centro de información, estadística o de generación de informes con retornos de la inversión espectacular.

Según Inmon (1996), reconocido como el padre de la data warehousing, Data Warehouse se caracteriza por ser:

Integrado: Los datos almacenados en el Data Warehouse deben integrarse en una estructura consistente, por lo que las inconsistencias existentes entre los diversos sistemas operacionales deben ser eliminadas. La información suele estructurarse también en distintos niveles de detalle para adecuarse a las distintas necesidades de los usuarios.

Temático: Sólo los datos necesarios para el proceso de generación del conocimiento del negocio se integran desde el entorno operacional. Los datos se organizan por temas para facilitar su acceso y entendimiento por parte de los usuarios finales. Por ejemplo, todos los datos sobre clientes pueden ser consolidados en una única tabla del Data Warehouse. De esta forma, las peticiones de información sobre clientes serán más fáciles de responder dado que toda la información reside en el mismo lugar.

Histórico: El tiempo es parte implícita de la información contenida en un Data Warehouse. En los sistemas operacionales, los datos siempre reflejan el estado de la actividad del negocio en el momento presente. Por el contrario, la información almacenada en el Data Warehouse sirve, entre otras cosas, para realizar análisis de

tendencias. Por lo tanto, se carga con los distintos valores que toma una variable en el tiempo para permitir comparaciones.

No volátil: El almacén de información de un “Data Warehouse” existe para ser leído y no modificado. La información es por tanto permanente, significando la actualización del Data Warehouse la incorporación de los últimos valores que tomaron las distintas variables contenidas en él sin ningún tipo de acción sobre lo que ya existía.

1.6 Sistema Gestor de Base de Datos

El concepto de BD esta ligado a los gestores de Base de Datos. En este epígrafe se ejemplificarán los diferentes Gestores de Base de Datos y el estudio va a estar centrado en el Gestor de Base de Datos PostgreSQL.

Los gestores de Base de Datos, se define como el conjunto de programas que administran y gestionan la información contenida en una BD, como la creación y manipulación de las bases de datos, los mismo permiten definir una base de datos (especificar tipos, estructuras y restricciones de datos), construir la base de datos (guardar los datos en algún medio controlado por el mismo SGBD) y manipular la base de datos (realizar consultas, actualizarla, generar informes). [2]

1.6.1 Algunos Gestores de Bases de Datos

En la actualidad existen un gran número de SGBD, por eso a continuación se ejemplifica algunos de los SGBD existentes y se analiza el SGBD PostgreSQL.

Sistemas de Gestores de Base de Datos

- Microsoft Access.
- Oracle.
- MySQL.
- Microsoft SQL Server
- PostgreSQL

PostgreSQL

Es un sistema de gestión de bases de datos objeto-relacional (ORDBMS) basado en el proyecto POSTGRES, de la universidad de Berkeley. El director de este proyecto es el profesor Michael Stonebraker, y fue patrocinado por Defense Advanced Research Projects Agency (DARPA), el Army Research Office (ARO), el National Science Foundation (NSF), y ESL, Inc. PostgreSQL es una derivación libre (OpenSource) de este proyecto, y utiliza el lenguaje SQL92/SQL9. Fue el pionero en muchos de los conceptos existentes en el sistema objeto-relacional actual, incluido, más tarde en otros sistemas de gestión comerciales. PostgreSQL es un sistema objeto-relacional, ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. A pesar de esto, PostgreSQL no es un sistema de gestión de bases de datos puramente orientado a objetos. [10] Con más de una década de desarrollo tras él, PostgreSQL es el gestor de BD de código abierto más avanzado hoy día, ofreciendo control de concurrencia multiversión, soportando casi toda la sintaxis SQL (incluyendo subconsultas, transacciones, y tipos y funciones definidas por el usuario), contando también con un amplio conjunto de enlaces con lenguajes de programación como pueden ser C, C++, Java, Perl, Python y otros. [30]

Como muchos otros proyectos de código abierto, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo. Dicha comunidad es denominada PGDG (PostgreSQL Global Development Group) por sus siglas en inglés.

A continuación se enumeran las principales características de este gestor de bases de datos:

- Implementación del estándar SQL92/SQL99.
- Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo: fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP), cadenas de bits, entre otros. También permite la creación de tipos propios.
- Incorpora una estructura de datos array.

- Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes, entre otras.
- Permite la declaración de funciones propias, así como la definición de disparadores.
- Soporta el uso de índices, reglas y vistas.
- Incluye herencia entre tablas (aunque no entre objetos, ya que no existen), por lo que a este gestor de bases de datos se le incluye entre los gestores objeto- relacionales.
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.

Presenta avanzadas funcionalidades como son: consultas SQL declarativas, control de concurrencia multi-versión, soporte multi-usuario, transacciones, optimización de consultas, herencia, y arrays.

Presenta otras características denominadas ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad): [3]

Atomicidad (Indivisible), es la propiedad que asegura que la operación se ha realizado o no, y por lo tanto ante un fallo del sistema no puede quedar a medias.

Consistencia, es la propiedad que asegura que sólo se empieza aquello que se puede acabar. Por lo tanto se ejecutan aquellas operaciones que no van a romper la reglas y directrices de integridad de la base de datos.

Aislamiento, es la propiedad que asegura que una operación no puede afectar a otras. Esto asegura que dos transacciones sobre la misma información nunca generarán ningún tipo de error.

Durabilidad, es la propiedad que asegura que una vez realizada la operación, ésta persistirá y no se podrá deshacer aunque falle el sistema.

Además PostgreSQL es altamente extensible, soporta operadores, métodos de acceso y tipos de datos definidos por el usuario. También contiene Soporte SQL Comprensivo. Además soporta la especificación SQL99, incluye características avanzadas tales

como las uniones (joins) SQL92 y su integridad referencial es utilizada para garantizar la validez de los datos de la base de datos.

Algunos límites de PostgreSQL se incluyen a continuación:

- Máximo tamaño de base de datos ilimitado.
- Máximo tamaño de tabla 32 TB.
- Máximo tamaño de tupla 1.6 TB.
- Máximo tamaño de campo 1 GB.
- Máximo tuplas por tabla ilimitado.
- Máximo columnas por tabla 250 - 1600 dependiendo de los tipos de columnas.
- Máximo de índices por tabla ilimitado.

Ventajas:

- Por su arquitectura de diseño, escala muy bien al aumentar el número de CPUs y la cantidad de RAM.
- Soporta transacciones y desde la versión 7.0, claves ajenas (con comprobaciones de integridad referencial).
- Tiene buen soporte para triggers y procedimientos en el servidor.
- Soporta un subconjunto de SQL92. Además, tiene ciertas características orientadas a objetos.

Desventajas:

- Consume muchos recursos y carga el sistema.
- Límite del tamaño de cada fila de las tablas a 8k (se puede ampliar a 32k recompilando, pero con un coste añadido en el rendimiento).
- Menos funciones en PHP.

1.7 Sistemas de soporte a la toma de decisiones (DSS), buenas prácticas, proceso de desarrollo de software.

Para dar un correcto orden a la guía y que la misma logre elevar la portabilidad, se deben analizar las definiciones de soporte a la toma de decisiones, así como sus clasificaciones, buenas prácticas, y qué es un proceso de desarrollo de software.

1.7.1 Qué es un sistema de Soporte a la Toma de Decisiones.

Turban y Aronson definen un Sistema de Soporte a la Toma de Decisiones (DSS por sus siglas en inglés) como: “Sistema interactivo basado en computadora, que ayuda a utilizar datos y modelos para resolver problemas no estructurados”. Los sistemas inteligentes, específicamente los sistemas de soporte a la toma de decisiones, tienen como uno de sus objetivos realizar análisis complejos y descubrir tendencias en el ámbito de los negocios. La información obtenida de ellos, permite a los gerentes tomar decisiones estratégicas a un bajo costo y en poco tiempo.

Little define un DSS como un modelo basado en un conjunto de procedimientos para procesar datos y juicios que ayudan a los administradores a tomar decisiones. Los sistemas de soporte a la toma de decisiones ofrecen beneficios específicos a los administradores o directores de todos los niveles. Por otra parte Bonczek define un DSS como un sistema basado en computadoras con tres componentes claves: un sistema de lenguaje (un mecanismo para proveer comunicación entre el usuario y otros componentes del DSS), un sistema de conocimiento (repositorio de conocimiento sobre problemas, datos o procesos) y el sistema procesador de problemas (la unión entre los otros dos componentes).

1.7.2 Clasificación de los DSS.

Power propone un modelo expandido para categorizar los sistemas más comunes de soporte a la toma de decisiones:

DSS Basados en Comunicación.

Es un tipo de DSS que da énfasis a comunicaciones, colaboración y apoyo de la toma de decisiones compartida. Los DSS Basados en Comunicación tienen por lo menos una de las siguientes características:

- Habilita la comunicación entre los grupos de las personas.
- Facilita el compartimiento de la información.
- La colaboración de apoyos y coordinación entre las personas.
- Tareas de decisiones de grupo de soporte.

Las tareas de investigación de DSS basado en Comunicación incluye los pactos en los procesos y conocimiento del grupo, interfaces multiusuario, control concurrente, comunicación y coordinación dentro del grupo, el espacio de información compartido y el apoyo de un ambiente abierto que integra las aplicaciones del usuario existentes. Los Sistemas de Decisiones Basados en Comunicación a menudo son categorizados según la matriz tiempo/localización usando la distinción entre los mismos y diferentes tiempos, y entre los mismos y diferentes sitios. (Power)

DSS Basados en Datos.

Estos sistemas analizan grandes fondos de datos encontrados en los sistemas principales de la organización. Ellos apoyan la toma de decisiones permitiendo a los usuarios extraer la información útil que antes fue introducida como cantidades grandes de datos. Por esta razón, a menudo los datos del sistema de tratamiento de transacción (TPS) son recogidos en depósitos de datos. El tratamiento online analítico (OLAP) y la minería de datos pueden entonces utilizarse para analizar los datos. Las compañías comienzan a construir DSS Basados en datos para extraer datos del cliente de sus sitios Web. Más DSS Basados en Datos están dirigidos a administradores, al personal y también a los proveedores de producto/servicio. Se usa para hacer consulta a una base de datos o almacén de datos para buscar las respuestas específicas para los propósitos específicos. Se despliega vía a un sistema del marco principal, vínculos cliente/servidor, o vía web, por ejemplo: base de datos basadas en computadoras que tienen un sistema de pregunta para verificar (incluso la incorporación de datos para agregar el valor a los bancos de datos existentes). (Power)

DSS Basados en Documentos.

Son muy comunes, dirigidos a sectores específicos en una amplia base de grupos de usuarios. El propósito de cada uno de estos Sistemas de Soporte a la Toma de Decisiones es registrar páginas web y encontrar documentos que contengan un conjunto específico de palabras claves o condiciones de búsqueda. La tecnología

usada para preparar los DSS basados en documentos es vía web o sistema cliente/servidor. (Power)

DSS Basados en Conocimiento.

Estos pueden sugerir o recomendar acciones a los líderes de proyectos, utilizan técnicas que constan del conocimiento acerca del dominio del problema en particular, y constan también de habilidades de resolución de esos problemas. Estos DSS proveen técnicas especializadas en resolución de problemas mediante datos almacenados en forma de hechos, reglas, procedimientos u otras estructuras similares. Las bases del conocimiento en esta clasificación de DSS se encuentran en: Fuentes externas de datos, Data Warehouse, meta base de datos, entre otros. (Power)

DSS Basados en modelos.

Los DSS Basados en Modelos son, principalmente, sistemas independientes aislados de los sistemas de información principales de la organización, que se usan para crear algún tipo de modelo capaz de ayudar a la toma de decisiones. Es un complejo sistema que ayuda a analizar decisiones o a elegir entre diferentes opiniones. Estos son usados por administradores y personal del negocio, o personas que interactúan con la organización, para un número de propósitos a merced de cómo es el modelo establecido y planificado así como la decisión analizada. (Power)

Otras clasificaciones.

En los Sistemas de Soporte a la Decisión se aborda otro esquema para la clasificación, ellos son: [9]

- Sistemas de información gerencial (MIS por sus siglas en inglés):

Los sistemas de información gerencial (MIS, Management Information Systems), también llamados Sistemas de Información Administrativa (AIS) dan soporte a un espectro más amplio de tareas organizacionales.

- Sistemas de información ejecutiva (EIS por sus siglas en inglés):

Los sistemas de información ejecutiva (EIS, Executive Information System) son el tipo de DSS que más se suele emplear, ya que proveen a los gerentes de un acceso

sencillo a información interna y externa de su compañía, y que es relevante para sus factores claves de éxito. En la siguiente figura se muestra cómo después de ver el QUE de un indicador en un EIS, si se desea analizar se hace una petición de información al Data Warehouse por cierta dimensión y devuelve información para entender el POR QUE.

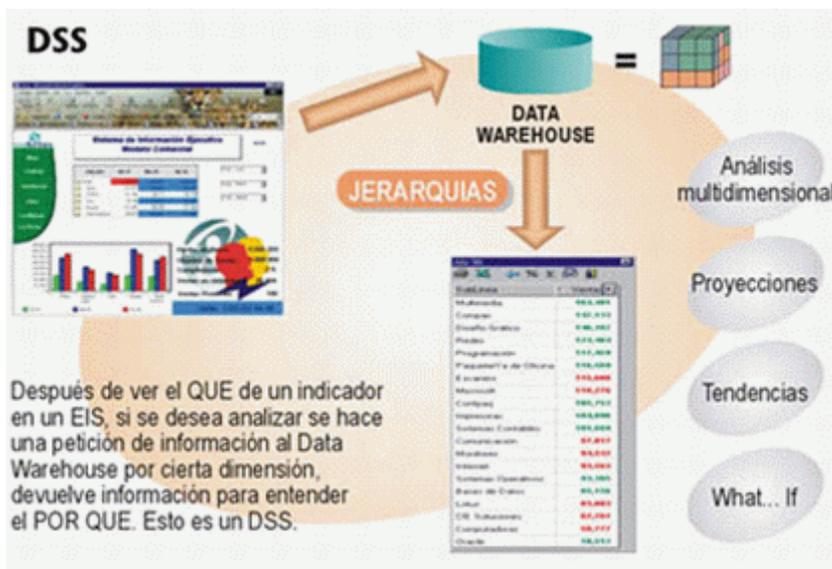


Figura 4: Contenido de un DSS.

- **Sistemas expertos basados en inteligencia artificial** (SSEE por sus siglas en inglés): Los sistemas expertos, también llamados sistemas basados en conocimiento, utilizan frecuentemente redes neuronales para simular el conocimiento de un experto y utilizarlo de forma efectiva para resolver un problema concreto.

- **Sistemas de apoyo a decisiones de grupo** (GDSS por sus siglas en inglés):

Un sistema de apoyo a decisiones en grupos (GDSS, Group Decision Support Systems) es "un sistema basado en computadoras que apoya a grupos de personas que tienen una tarea (u objetivo) común, y que sirve como interfaz con un entorno compartido". El supuesto en que se basa el GDSS es que si se mejoran las comunicaciones se pueden mejorar las decisiones.

Con las definiciones vistas anteriormente se llega a la conclusión de que DSS, ayuda a la toma de decisiones de los líderes de un proyecto o administradores a la hora de entrelazar datos y de un software accesible en un entorno potente, vigoroso que pueda dar soporte a la toma de decisiones, por lo tanto se define como un conjunto de

herramientas que permiten obtener de manera oportuna la información que se requiere mediante el proceso de la toma de decisiones que se desarrolla en un ambiente de incertidumbre.[9]

1.7.3 ¿Qué es una buena práctica?

Es una técnica, método, proceso, actividad, incentivo o reconocimiento, que es más efectivo que cualquier otro conocido, para entregar cierto resultado. [24]

1.7.4 Proceso de desarrollo de software

Se define como Proceso de desarrollo de software al conjunto ordenado de pasos a seguir para llegar a la solución de un problema u obtención de un producto, así como lograr la obtención de un producto software que resuelva un problema.[31]

1.6 Conclusiones

En este capítulo se han analizado definiciones fundamentales para llevar a cabo un correcto orden en la investigación, con el objetivo de crear una guía que obtenga la adaptabilidad, instalabilidad y coexistencia de las soluciones software de apoyo a la toma de decisiones basadas en Data Warehouse.

Se proporcionó información que permite comprender la necesidad de elevar la portabilidad de las soluciones software desarrollado en los centros de producción de la Universidad de Ciencias Informáticas.

Se arribó a la conclusión de que la mejor solución a este problema es analizar la base de dato Data Warehouse. También se estudió las principales características del SGBD PostgreSQL y por qué es importante la toma de decisiones.

Se determinó que el modo más factible para crear una guía que obtenga la adaptabilidad, instalabilidad y coexistencia de las soluciones software, es brindar buenas prácticas que eleven la portabilidad de dichos software, por lo tanto se sentaron las bases teóricas que se deben tener en cuenta para el desarrollo del capítulo dos.

2. Capítulo 2: Propuesta de solución

En el presente capítulo se presentará la Propuesta de Solución para resolver el problema científico de la investigación cumpliéndose el objetivo trazado al inicio del capítulo uno en correspondencia con la fundamentación teórica. La solución que se propone se basa en una guía de actividades para obtener la adaptabilidad, instalabilidad y coexistencia y, de esta forma tomar correctas decisiones técnicas.

Con el resultado de este capítulo se tendrá descrito durante el Proceso de Desarrollo de las Soluciones Software de ayuda a la toma de decisiones, que se desarrollan en la línea de Soluciones de Almacenes de Datos e Inteligencia de Negocio, del Centro DATEC, cómo va a estar estructurada la guía para elevar la portabilidad.

2.1 Algunas características de la Línea de Soluciones de Almacenes de Datos e Inteligencia de Negocio en DATEC

En un estudio realizado en la Línea de Soluciones de Almacenes de Datos e Inteligencia de Negocio en el centro de DATEC, se obtuvo datos de las herramientas utilizadas por los mismos. A continuación se muestra una tabla donde se observan los datos.

Características de la Línea de Soluciones	Herramientas, lenguajes, metodologías
Control de versiones	SVN
Sistema operativo	Linux
Herramienta de modelado	Visual Paradigm
Gestión de proyecto	Remain
Metodología	Kimball
Seguimiento de errores	Redmine
Lenguajes de programación	Java, PHP, SQLServer

Tabla 1: Herramientas, lenguajes y metodología

2.2 Factores que afectan la portabilidad.

Luego de un detallado estudio y análisis de las principales actividades que se llevan a cabo en DATEC y, del análisis que se mostró en el capítulo uno sobre los modelos de calidad y los estándares, se llega a la conclusión de que existen diversos factores que afectan a las soluciones software a la hora de adaptarlos e instalarlos en otros ambientes específicos, así como el hecho de que los mismos coexistan con otros recursos. A continuación se mencionan y se explican cada uno de ellos.

Garantizar la visualización de datos

La visualización de datos o de información es el proceso de interpretar en términos visuales, imágenes o textos que están sostenidos por la computadora para ampliar conocimientos que anteriormente se encontraban en un formato abstracto, por lo tanto el usuario puede percibir la calidad del software desde el primer instante en que interactúa con la aplicación, la cual utiliza un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz.

El manejo de las funcionalidades de las soluciones software debe ser lo más intuitivo posible, de manera que sean muy claras las posibles acciones a llevar a cabo y la manera de hacerlas, de ahí que las soluciones software presentan dificultades para ajustarse a un determinado entorno, como: la variabilidad que presenta el formato de datos para personalizar una gráfica, la apariencia que se desee, así como, las deformaciones sobre la lectura de formato de datos- directos en cualquier interfaz existente, incluso la nitidez de las imágenes, entonces la compatibilidad de aplicaciones en ciertos sistemas operativos no es la misma.

Priorizar la exportación o importación de los datos

Los datos son la expresión general que describe los objetos con los cuales opera el programa, por lo tanto optimizar el acceso, la disponibilidad y la reutilización de los

mismos, así como registrar y documentar forman parte del papel protagónico en una organización.

La exportación o importación de datos en las soluciones software presenta dificultades debido a que se introducen datos manualmente desde un archivo, consecuencia de la ausencia de un sistema que pueda suministrar las entradas de datos para estos campos, incluso limita al usuario que pueda importar el mismo archivo varias veces., además la única limitación de tamaño de un archivo será la máxima que permita el soporte físico utilizarlo para su transporte. También la salida de los datos en ocasiones sufren cambios o distorsiones al mostrarse en otros navegadores, por tanto no publican un documento de especificación que describa exactamente como debe ser la conversión de los datos, y que se puede usar para determinar si un programa concreto trata un formato de archivo particular correctamente o no.

Características y condiciones del ambiente organizacional

Se deben tener en cuenta las condiciones que existan en el ambiente organizacional en el que será instalada la aplicación, para garantizar que al ser instalada ésta se adapte al entorno con mayor facilidad.

Garantizar que el software sea multiplataforma

El Software Multi-Plataforma consiste en instalar la aplicación sólo en un equipo, el servidor, y todos los clientes entrarán al sistema por medio de una interfaz. , donde se tenga la posibilidad de acceder a cualquier tipo de Sistema Operativo (Linux, Windows, Mac OS, entre otros).

Cuando se elabora una aplicación y esta solo puede ser instalada en una computadora que posea la misma plataforma que donde fue creada. Por ejemplo la plataforma IBM-PC, x86, IA64 o AMD64 y SPARC. La dependencia de una plataforma hardware/software concreta supone una molesta atadura para el usuario y crean ligaduras que dificultan el paso a otra arquitectura diferente.

Disponibilidad de una interfaz de instalación

La interfaz de instalación comprende todos los puntos de contacto entre el usuario y la máquina, por tanto es el medio con que el usuario puede comunicarse con la computadora, y normalmente suelen ser fáciles de entender y fáciles de accionar.

Unas de las dificultades que existen en las soluciones software es que no suministra un entorno amigable de la interfaz de instalación de usuario, debido a que las funciones no tienen una imagen representativa que facilite el acceso. Además existe incoherencia de operación con los módulos, por lo que el proceso de aprendizaje para los usuarios se tornaría lenta y por tanto se requiere de personal especializado en conocimientos informáticos.

Documentar manuales de instalación

Los manuales son los que contienen las instrucciones para que nos sea posible realizar las tareas que pretendemos y que convierten a los ordenadores en máquinas que responden a determinadas funciones que se requieren de ellas.

La ausencia de los manuales de instalación limita a que el usuario reciba referencias de las particularidades del equipo o del sistema y la forma de utilizarlo, además de que no pueda consultar cuando se haya olvidado de algún aspecto o se haya presentado alguna duda.

Conocimiento de las dependencias de instalación con otro software

El software de dependencia es una aplicación requerida por otro programa para poder funcionar correctamente. Por ello se dice que dicho programa depende de tal aplicación.

Algunas formas de instalación de software, como el uso de instaladores o la compilación del código fuente, no buscan ni descargan automáticamente las dependencias del programa siendo esto un verdadero problema para el usuario, ya que descarga e instala manualmente las aplicaciones necesarias.

Identificar requerimientos mínimos del hardware y software

Los componentes del hardware se pueden clasificar en básicos y complementarios, los básicos son aquellos necesarios para otorgar la funcionalidad mínima a una computadora y los complementarios son utilizados para realizar funciones específicas, no precisamente para el funcionamiento de la computadora, en contraposición a los componentes físicos del sistema, el conjunto de los componentes lógicos para hacer posible la realización de tareas específicas, son llamados software y se refieren al equipamiento lógico o soporte lógico de una máquina.

Características mínimas que debe poseer la PC para soportar la aplicación. (Memoria RAM, velocidad del micro, capacidad). Requerimientos de rendimiento, de interfaz. ¿Cuántas veces ha pasado que el servidor ha quedado muy chico o solamente se tiene la capacidad suficiente para un año? Muchas veces se pone la capacidad al mínimo, luego se tiene las PCs con 256 Mb de memoria, en las que el usuario tendrá que correr el sistema operativo, su correo, las noticias, la información de la compañía, la aplicación y los otros programas que el usuario posee, y entonces aparece el problema de que no se tiene suficiente espacio para soportar todas las aplicaciones al mismo tiempo.

Establecer el uso mínimo de recursos.

En una organización cuya infraestructura de servicios presenta componentes de hardware, software y base de datos muy versátiles, los entornos operacionales deben coexistir con las diversas tecnologías (libres y propietarias) o recursos a fin de garantizar el correcto funcionamiento y las operaciones adecuadas de sus servicios.

En varias ocasiones no se tiene en cuenta las modificaciones del hardware, lo cual puede provocar fallos en el sistema y lentitud en las operaciones ya que la PC tiene que ocuparse de una gran cantidad de solicitudes diferentes, y cada una de ellas consume parte de la energía de procesamiento de la PC.

A continuación se observa en una tabla de los factores teniendo en cuenta la subcaracterística que afectan.

Factores Potenciales	Subcaracterísticas de portabilidad
-----------------------------	---

Garantizar la visualización de los datos	Adaptabilidad
Priorizar la exportación o importación de los datos	
Características y condiciones del ambiente organizacional	
Garantizar que el software sea multiplataforma	
Disponibilidad de una interfaz de instalación	Instalabilidad
Documentar manuales de instalación	
Conocimiento de las dependencias de instalación con otro software	
Identificar requerimientos mínimos del hardware y software	
Establecer el uso mínimo de recursos	Coexistencia

Tabla 2: Los factores que afectan las subcaracterísticas de portabilidad

2.4 Propuesta de la guía de solución

Teniendo en cuenta el proceso de desarrollo del software que se lleva a cabo en la Línea de Soluciones de Almacenes de Datos e Inteligencia de Negocio en el centro de DATEC, se llegó a la creación de una guía para obtener la adaptabilidad, instalabilidad y coexistencia de las soluciones software al apoyo de la toma de decisiones. Esta cuenta de seis fases:

Fase 1: Estudio Preliminar.

Fase 2: Requerimientos y gestión del proyecto

Fase 3: Arquitectura

Fase 4: Diseño e Implementación.

Fase 5: Implantación y Operaciones.

Fase 6: Soporte.

Cada fase cuenta con una descripción de las principales entradas que necesita para su realización, el principal objetivo que persigue, las actividades que se deben realizar para obtener la portabilidad, y también se incluyen las principales salidas que generan. En la figura 6 se representa el proceso de desarrollo de las soluciones software en la Línea de Soluciones de Almacenes de Datos e Inteligencia de Negocio en el centro de DATEC.

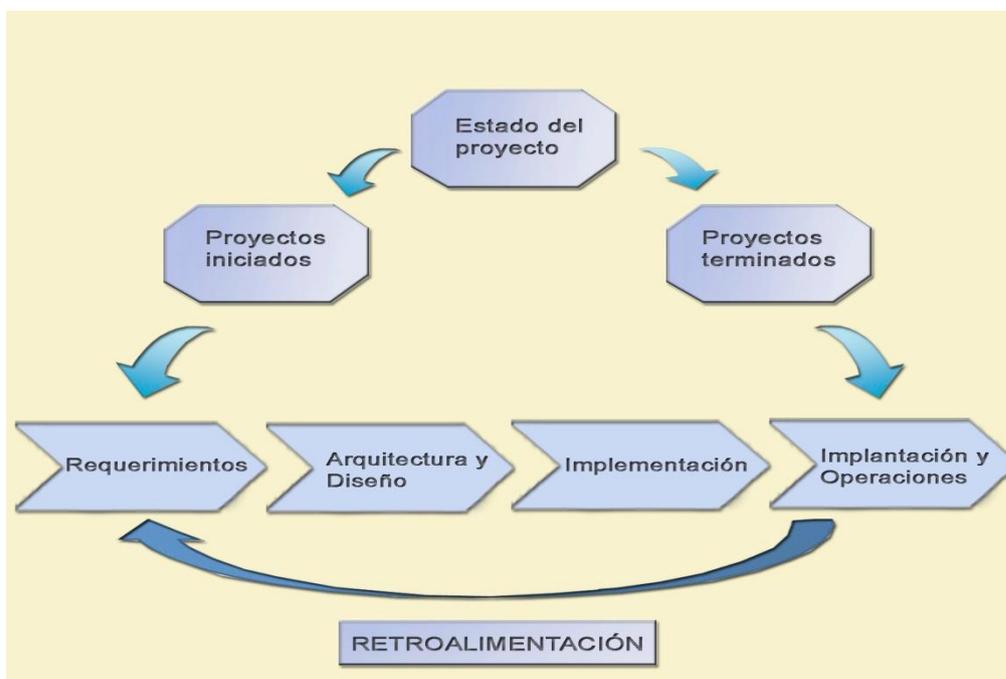


Figura 5: Proceso de desarrollo de las soluciones software.

A continuación se realiza la descripción de cómo quedó estructurada cada una de las fases propuestas en la guía de solución.

Fase 1: Estudio Preliminar

En la fase de Estudio Preliminar se llevan a cabo las actividades relacionadas con la planeación del proyecto a un alto nivel y la legalización del mismo. En esta fase se realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcance del proyecto y dentro de este lograr la portabilidad. Además se necesita de la elaboración de un glosario de términos, debido a las desigualdades de conceptos empleados entre los desarrolladores, siendo esto una principal importancia para la comunicación y el consenso en los mismos y sus definiciones.

Objetivos:

- Planeación del proyecto, su legalización y alcance del mismo.
- Obtener un acuerdo acerca de los conceptos y sus definiciones.

Entradas:

- Trazar el alcance del proyecto.
- Comunicaciones entre los miembros del equipo y el cliente.
- Definiciones propuestas.

Actividades:

- Definir las actividades para la organización del proyecto y su alcance.
- Obtener consenso sobre las definiciones.

Descripción de las actividades:

Definir las actividades para la organización del proyecto y su alcance: En esta actividad se establece una descripción textual de las principales tareas que desarrollará el proyecto, su organización, así como establecer a cada miembro del equipo el rol que le pertenece y las orientaciones de las tareas a cumplir.

Obtener consenso sobre las definiciones: En esta actividad el papel importante está dada en la igualdad de pensamiento de todo el equipo, por tanto, cada miembro de dicho equipo, guiados por el líder de proyecto llegarán a un acuerdo común sobre las definiciones del problema que deben desarrollar.

Salidas:

- Una documento elaborado para cumplir necesidades específicas del alcance del proyecto.
- Estructuración del proyecto.
- Glosario de términos.

Fase 2: Requerimientos y gestión del proyecto

Durante la fase de requerimientos se definen de manera detallada las funcionalidades del sistema, reglas, hechos, requerimientos funcionales y no funcionales. En esta fase se debe tener en cuenta todos los factores estudiados en el epígrafe anterior: garantizar la visualización de los datos, priorizar la exportación o importación de los datos, características y condiciones del ambiente organizacional, garantizar que el software sea multiplataforma, disponibilidad de una interfaz de instalación, documentar manuales de instalación, conocimiento de las dependencias de instalación con otro software, identificar requerimientos mínimos del hardware y software y, establecer el uso mínimo de recursos.

Objetivos:

- Definición de los requerimientos de usuario.
- Definir el ámbito del sistema enfocada a las necesidades y metas del usuario.
- Recoger la información sobre el funcionamiento de los procesos actuales.

Entradas

- Plan de Gestión de Requisitos.
- Cronograma de Entrevistas.
- Resumen y Resultados de las Entrevistas.
- Cuestionarios.

Actividades:

- Especificar las condiciones adecuadas para un ambiente red íntegra.
 - Topología de red.
 - Protocolo de comunicación.
- Revisar las dependencias que presenta la plataforma en la que va a ejecutarse la solución software.
- Refinar los requisitos de hardware y software para la instalación y ejecución de la solución software.
- Efectuar entrevistas y encuestas para determinar los requisitos del sistema al interactuar con el usuario.
- Documentar las recopilaciones de las entrevistas y encuestas.

Salidas:

- Especificación de Requisitos
- Modelo de integración de datos
- Diseño Conceptual
- Plan de Pruebas
- Diseño de Casos de Prueba

Fase3: Arquitectura y Diseño

Esta fase, es la encargada de realizar una serie de actividades con el objetivo de crear las bases sobre las cuales se desarrollará e implementará las soluciones software del proyecto. Durante el desarrollo de esta fase se comienza la elaboración de una serie de documentos a partir de la información obtenida en la fase de requisitos y se le da respuesta a los diferentes factores potenciales como: características y condiciones del ambiente organizacional, garantizar que el software sea multiplataforma, conocimiento de dependencia de instalación con otros software, identificar requerimientos mínimos del hardware y software y, establecer el uso mínimo de recursos.

Objetivos

- Diseño Técnico de la Arquitectura.

Entrada

- Modelo de integración de datos.
- Informe Levantamiento de Arquitectura de Información.

Actividades

- Analizar los protocolos de interacción y comunicación.
- Definir los patrones y arquitecturas más apropiados para que las soluciones software puedan ser instaladas en cualquier plataforma.
- Asegurar el mínimo de recursos que utilizará la solución software para evitar que existan problemas a la hora de coexistir en el ambiente real.
- Seleccionar la Topología de Red adecuada.
- Identificar los requisitos de hardware y software para la instalación y ejecución de la solución software.

Descripción de las actividades:

Analizar los protocolos de interacción y comunicación: En esta actividad se debe determinar el protocolo bajo el cual se pueda realizar una mejor comunicación entre la aplicación y las bases de datos.

- Se sugiere utilizar el protocolo de comunicación TCP/IP (Protocolo de Control de Transmisión (TCP) y Protocolo de Internet (IP)) ya que este sirve para enlazar computadoras que utilizan diferentes sistemas operativos, incluyendo PC, minicomputadoras y computadoras centrales sobre redes de área local (LAN) y área extensa (WAN).
- Este tiene un grado muy elevado de fiabilidad, es adecuado para redes grandes y medianas, así como en redes empresariales. Proporciona una abstracción del medio haciendo posible el intercambio de información entre medios desiguales y tecnologías que un principio eran incompatibles.

Definir los patrones y arquitecturas más apropiados para que las soluciones software puedan ser instaladas en cualquier plataforma: Es aquí donde entra a jugar un papel importante la arquitectura por capas. Ver anexo 2.

- Esta arquitectura favorece a la adaptabilidad, satisface los requerimientos definidos por los usuarios del sistema y otorga algunas ventajas como es el caso de crear a través de patrones de diseño una aplicación adaptable a diferentes plataformas sin límites de dependencia, tal es el caso de:
 - **Abstract Factory** que proporciona una interfaz para crear familias de objetos que dependen entre sí, sin especificar sus clases concretas.
 - **Bridge o Puente** que desvincula una abstracción de su implementación, de manera que ambas puedan variar de forma independiente.

Asegurar el mínimo de recursos que utilizará la solución software para evitar que existan problemas a la hora de coexistir en el ambiente real: En la presente actividad se recomienda considerar el mínimo de recursos que utilizará la solución software para evitar que existan problemas a la hora que esta sea instalada en el ambiente real.

- De darse la situación en la que dos o más procesos lean o escriban en un área de memoria compartida se recomienda impedir que más de un proceso acceda simultáneamente a las variables compartidas y esto se logra garantizando la exclusión mutua. Para solucionar el problema de la exclusión mutua se tiene en cuenta tres tipos de soluciones:

1. Soluciones hardware:

Las soluciones hardware van estar divididas en dos tipos:

- Optimistas: donde se considera que lo más probable es que no haya conflictos, y si los hay sea en número reducido, por lo que permiten cualquier acceso a la variable compartida. En caso de conflicto, mantienen la integridad del sistema descartando las actualizaciones. Ejemplo de ello es el método comparar e intercambiar.
- Pesimistas: donde se bloquea todo aquello que pueda interferir, se actualizan la variable y se desbloquea lo bloqueado al principio. Ejemplo de ello es deshabilitar interrupciones y habilitar interrupciones, Test and Set (Compartir y Fijar).

2. Soluciones de software:

- Esta solución consiste en dejar la responsabilidad a los procesos que se deseen ejecutar concurrentemente, de esta manera los procesos deben coordinarse unos con otros para cumplir la exclusión mutua sin ayuda alguna, aunque están propensas a errores y a una fuerte sobrecarga de proceso (Algunos ejemplos: Algoritmo de Dekker y Algoritmo de Peterson).

3. Soluciones aportadas por el Sistema Operativo.

- Se utilizan los semáforos ya que resultan adecuados cuando hay que proteger un recurso que puede compartir varios procesos.
- Los mensajes también son una solución del sistema operativo que permite la sincronización de procesos y la comunicación entre ambos. Estos son una cadena de texto transmitida de un proceso que envía el mensaje a un receptor (proceso receptor). Un mensaje puede tener código, texto, datos. [26]

Seleccionar la topología de red adecuada: En esta actividad se debe tener en cuenta la topología de red existente de forma tal, que cuando se instale la solución no sobrecargue el servidor y evite la caída de los servicios.

Para establecer la topología de red se deben tener en cuenta las necesidades y características de la empresa, una selección óptima de la misma garantiza un buen funcionamiento e impide que se formen cuellos de botella. Existen diferentes tipos de red, entre ellas: (Ver anexo 4)

➤ **Topología de estrella (star):**

En una topología de estrella, las computadoras en la red se conectan a un dispositivo central conocido como concentrador (hub en inglés) o a un conmutador de paquetes (switch en inglés). Cada computadora se conecta con su propio cable (típicamente par trenzado) a un puerto del hub o switch. Este tipo de red sigue siendo pasiva, utilizando un método basado en contención, las computadoras escuchan el cable y contienden por un tiempo de transmisión. La desventaja de esta topología es en la centralización de la comunicación, ya que si el hub falla, toda la red se cae.

➤ **Topología de anillo (ring):**

Una topología de anillo conecta los dispositivos de red uno tras otro sobre el cable en un círculo físico. La topología de anillo mueve información sobre el cable en una dirección y es considerada como una topología activa.

➤ **Topología de malla (mesh):**

La topología de malla (mesh) utiliza conexiones redundantes entre los dispositivos de la red así como una estrategia de tolerancia a fallas. Cada dispositivo en la red está conectado a todos los demás (todos conectados con todos). Este tipo de tecnología requiere mucho cable (cuando se utiliza el cable como medio, pero puede ser inalámbrico también). Pero debido a la redundancia, la red puede seguir operando si una conexión se rompe. Las redes de malla, obviamente, son más difíciles y caras para instalar que las otras topologías de red debido al gran número de conexiones requeridas. [28]

➤ **La topología Árbol jerárquico:** es la más empleada y muy útil para implementar en grandes empresas, pues la señal puede viajar más lejos. Sin embargo su implementación es difícil de configurar y si se cae la rama principal toda la red lo hará.[28]

Identificar los requisitos de hardware y software para la instalación y ejecución de la solución software: Durante esta actividad se documentan los requisitos para la instalación y posteriormente la ejecución de la solución software, para lograr de esta manera que el rendimiento sea eficaz y rápido. Se recomienda en este caso tener presente los siguientes ejemplos, el servidor donde se instalará la Base de Datos del sistema debe cumplir con los siguientes requisitos (puede ser el mismo donde estará la aplicación):

- SO: GNU/Linux preferentemente Ubuntu GNU/Linux 8.04 o superior.
- Debian 4 GNU/Linux o superior.
- PostgreSQL versión 8.3 o superior.
- PGAdmin III o algún administrador para PostgreSQL.

Para lograr instalar PostgreSQL en Linux se deben tener en cuenta los siguientes requerimientos mínimos:

- 8 Mb de RAM.
- 30 Mb de espacio en disco para el código fuente.
- 5 Mb de espacio en disco para la instalación de los ejecutables.
- 1 Mb extra para las base de datos básicas.
- 3 Mb de espacio en disco rígido para el trabajo con el código fuente.

Para instalar PostgreSQL en Windows se debe tener en cuenta:

- Que el Sistema Operativo (SO) sea estable y funcione bien, que su versión sea 8.2 o superior, ya que en esta se corrigen errores de código que tienen las versiones anteriores y así no se pone en riesgo la estabilidad del código.
- Memoria RAM de 128 MB como mínimo para Windows 2000 y de 256 MB para XP.
- Capacidad de 512 MB.

Salida

- Documento de Arquitectura del Sistema.
- Arquitectura de información.

Fase 4: Implementación

En esta fase con base al diseño de conocimiento ya definido, se inicia la construcción del Sistema e implementación en el lenguaje que se definió anteriormente. Además se realizan, los procesos de extracción, carga y limpieza de datos para el almacén, se implementa el almacén y la aplicación Inteligent Bussinnes (BI). Aquí están implícitos los factores como: garantizar la visualización de los datos, disponibilidad de una interfaz de instalación, priorizar la exportación o importación de los datos.

Objetivos

- Análisis de Fuentes de Datos.
- Especificación y Desarrollo de Aplicaciones de Usuario Final.

Entrada

- Diseño de pruebas de integración.
- Actualización de plan de pruebas.
- Diseño de Casos de Pruebas.

Actividades

- Garantizar la existencia de estándares de conversión para la entrada y salida de datos.
- Lograr que las dependencias de las soluciones software se descarguen automáticamente durante el proceso de instalación.
- Desarrollar un asistente de instalación que comprenda pasos automatizados.
- Seleccionar un lenguaje de programación acorde a las necesidades visuales de datos.

Descripción de las actividades

Garantizar la existencia de estándares de conversión para la entrada y salida de datos: En esta actividad se garantiza que se desarrolle en la herramienta la posibilidad de importar datos de aplicaciones similares con diferentes estándares de formato o extensiones. Además se debe publicar y especificar claramente este tipo de información en la ayuda que se crea para servir de apoyo a la aplicación.

- Esta actividad garantiza que el procedimiento para la transferencia de datos antiguos acepte la misma estructura de entrada de datos de las soluciones software y también que al mostrarse los datos en un navegador no sufran cambios o distorsiones en las imágenes, por lo que:
- Se recomienda estandarizar la salida de los datos mediante la validación del código de la web en base a los estándares del W3C, que elimina la ocurrencia de errores y se agiliza esta tarea que de hacerse manualmente resultaría muy tediosa

Lograr que las dependencias de las soluciones software se descarguen automáticamente durante el proceso de instalación: El objetivo de esta actividad es lograr que durante el transcurso del proceso de instalación del software aparezcan de manera adjunta todos los programas asociados o de los que la aplicación vaya a depender para su ejecución posteriormente. De esta forma la instalación se tornará menos engorrosa. Además, se garantiza que no exista la necesidad de descargar dichas dependencias y que disminuya notablemente el tiempo de la instalación.

Desarrollar un asistente de instalación que comprenda pasos automatizados: Con esta actividad se procura que cuando el usuario interactúe con la interfaz de instalación tome la menor cantidad posible de decisiones y, así se logra que la interfaz sea intuitiva y que los usuarios no necesiten conocimientos informáticos especializados. Por lo tanto el tiempo de instalación no será extenso.

- Se recomienda que en las ventanas de asistente existan imágenes representativas, y además, que las selecciones óptimas por defecto aparezcan automáticamente seleccionadas.

Seleccionar un lenguaje de programación acorde a las necesidades visuales de datos: Con el desarrollo de esta actividad se logra una correcta selección del lenguaje de programación ideal para el análisis y procesamientos de datos, lo cual genera una velocidad agradable para mostrar las cosas.

- En este caso se recomienda como lenguaje de programación Java.

Salida

- Manual de Usuario

- Especificaciones del usuario final
- Diseño de pruebas de integración
- Código fuente.
- Plan de pruebas

Fase 5: Implantación y Operaciones

Durante esta fase se procede a implantar la solución y a la capacitación al cliente en las herramientas a utilizar y se le deja todo listo para la explotación del almacén. Se hacen las pruebas de calidad a cada una de las unidades implementadas, así como pruebas de integración y del sistema de manera general, para la comprobación de la fiabilidad del sistema.

Objetivos

- Pruebas
- Mantenimiento
- Crecimiento
- Identificar los defectos y corregirlos antes de la instalación

Entrada

- Guía de Implantación

Actividad

- Verificar que se satisfagan los requerimientos

Descripción de la actividad

Verificar que se satisfagan los requerimientos: En esta actividad se evalúan cada uno de los requerimientos que fueron especificados en el inicio del proceso de desarrollo. Se recomienda la elaboración de las pruebas correspondientes para comprobar el cumplimiento de los mismos.

Salida

- Actualización de plan de pruebas
- Pruebas de Unidad e integración
- Pruebas del sistema.

Fase 6: Soporte

La fase de soporte es el período donde la solución está implantada y los usuarios finales están utilizándolos y se generan una serie de no conformidades de las mismas.

Entrada

- Desarrollar las Condiciones de Soporte.

Actividad

- Documentar los cambios que se originen durante el soporte de la solución software.

Descripción de la actividad

Documentar los cambios que se originen durante el soporte de la solución software: Los cambios son inevitables, pero para ello es necesario evaluar si éstos son necesarios y rastrear su impacto. Monitorear los cambios dentro del proceso iterativo de desarrollo, mantener toda la organización informada sobre los avances y resultados del programa, son aspectos importantes para el proceso de soporte.

- Por tales motivos, en esta actividad se recomienda realizar un documento donde se registre las no conformidades del sistema, para preparar la retroalimentación del proceso de desarrollo del software.

A continuación la tabla muestra cada factor que afecta a las subcaracterísticas de portabilidad ubicados en las fases de la guía, donde se propusieron buenas prácticas para elevar la portabilidad.

Factores potenciales	Subcaracterísticas de Portabilidad			Fases del ciclo de vida del software						
	A	I	C	EP	R	A y D	I	I y O	S	
Garantizar la visualización de los datos	x				x		x			
Priorizar la exportación o importación de los datos	x				x		x			
Características y condiciones del ambiente organizacional	x				x	x				
Garantizar que el software sea multiplataforma	x				x	x				
Disponibilidad de una interfaz de instalación		x			x		x			
Documentar manuales de instalación		x			x					
Conocimiento de las dependencias instalación con otro software		x			x	x				
Identificar requerimientos mínimos del hardware y software		x			x	x				
Establecer el uso mínimo de recursos			x		x	x				

Tabla 3: Tabla de factores ubicados en las fases de la guía.

2.5 Conclusiones

En el capítulo se realizó la definición de las fases por la que está compuesta la guía, las actividades a realizar en cada una de las fases definidas, donde se cumple con los objetivos trazados en el comienzo de la investigación. Se definió y justificó la selección de los factores que afectan en gran medida las soluciones software para elevar su portabilidad y durante las fases del ciclo de vida de la soluciones se toman buenas prácticas para dar solución a estos inconvenientes informáticos.

3. Capítulo 3: Validación de la propuesta

El objetivo de la investigación es realizar una guía de buenas prácticas para obtener la adaptabilidad, instalabilidad y coexistencia de las soluciones software y así elevar la portabilidad de dichas soluciones basadas en BD Data Warehouse; para dar cumplimiento a dicho objetivo se hace necesario en el desarrollo de este capítulo realizar la validación de la propuesta de solución descrita en el capítulo anterior mediante la valoración de un conjunto de expertos para que brinden sus criterios y sugerencias. Con este fin se utilizan técnicas propuestas por el método Delphi, basado en el criterio de expertos. El capítulo está estructurado de manera que brinde las características fundamentales del método utilizado y la explicación detallada de sus fases.

3.1 Métodos de Validación

Existen varios métodos de validación, entre los cuales se puede encontrar: [18]

- Comparación de los resultados de salida del modelo con los del sistema real.
- Test de Turing.
- Validación de comportamientos en casos extremos.
- Método Delphi.

A continuación se presentan breves características de cada uno de los métodos anteriormente mencionados:

3.1.2 Comparación de los Resultados de Salida del Modelo con los del Sistema Real:

Éste método presenta 3 características principales:

- El sistema real debe existir.
- Se comparan las salidas del modelo con las del sistema, mediante algún método estadístico.

- Dado que la mayoría de los procesos de salida no son estacionarios y si autocorrelados, estos test no son directamente aplicables. [18]

Test de Turing:

Alan Turing sugirió este método como un test de inteligencia artificial. En este test, a un experto, o grupo de expertos, se le presentan resúmenes o informes de resultados de ejecución del sistema y del modelo, a los que se les ha dado el mismo formato. Estos informes se reparten aleatoriamente a los ingenieros y administradores del sistema, para ver si son capaces de discernir cuales son los reales del sistema y cuales la imitación resultado de la simulación. Si los expertos no son capaces de distinguir entre ambos, se puede concluir que no hay evidencias para considerar inadecuado al modelo. Si descubren diferencias las respuestas sobre lo que encuentran inconsistente se puede utilizar para realizar mejoras en el modelo.

Validación de Comportamientos en Casos Extremos:

Ocasionalmente se puede observar el comportamiento del sistema bajo condiciones extremas. Esta es una situación ideal para recoger datos de las medidas de ejecución del sistema real de forma que luego se puedan comparar con los resultados de la simulación, una vez que se ejecute el modelo bajo situaciones similares. También es posible que los expertos del sistema puedan predecir el comportamiento del sistema bajo condiciones extremas y utilizar estas predicciones para validar el modelo. [18]

Características generales del método Delphi:

El primer estudio Delphi fue realizado en 1950 por la Rand Corporation para la fuerza aérea de EE.UU. y se le dio el nombre de "Proyecto Delphi". El objetivo de este estudio fue obtener el mayor consenso posible en la opinión de un grupo de expertos por medio de una serie de cuestionados intensivos, a los cuales se les intercalaba una retroalimentación controlada.

En el estudio realizado varios de los autores recomiendan no darle una definición como tal al método Delphi, por lo que se llega a la conclusión que la mejor forma es brindar un análisis de la descripción general de sus características.

El método Delphi es una técnica grupal de análisis de opinión, parte de una suposición fundamental y de que el criterio de un individuo particular es menos fiable que el de un grupo de personas en igualdad de condiciones, en general utiliza e investiga la opinión de expertos.

El principio de este método es la inteligencia colectiva tratando de lograr un consenso de opiniones expresadas individualmente por un grupo de personas seleccionadas cuidadosamente como expertos calificados en torno al tema, por medio de la iteración sucesiva de un cuestionario retroalimentado de los resultados promedio de la ronda anterior, aplicando cálculos estadísticos.

Las principales características del método están dadas por el anonimato de los participantes (excepto el o los investigadores), iteración (manejar tantas rondas como sean necesarias), retroalimentación controlada, respuesta de grupo en forma estadística y justificación de respuestas.

Existen tres etapas o fases esenciales en la aplicación del método:

- Fase preliminar. Se delimita el contexto, los objetivos, el diseño, los elementos básicos del trabajo y la selección de los expertos.
- Fase exploratoria. Elaboración y aplicación de los cuestionarios según sucesivas vueltas, de tal forma que con las respuestas más comunes de la primera se confecciona la siguiente.
- Fase final. Análisis estadísticos y presentación de la información.

Una vez estudiados los métodos de validación se ha llegado a la conclusión de utilizar en la investigación el método Delphi, ya que mediante éste los expertos deben predecir los resultados a alcanzar con la propuesta elaborada. Además es muy exacto para obtener información sobre el futuro. El procesamiento estadístico y matemático de la información es la característica más importante del método que lo diferencia del resto de los métodos, ya que la decisión final que se toma es un criterio fuertemente avalado por la experiencia y conocimiento del colectivo de expertos consultados. Metodológicamente se hace imprescindible, a la hora de aplicar este método, tener en cuenta pasos necesarios para la aplicación del mismo al problema en cuestión:

1. Selección del grupo de expertos a entrevistar
2. Elaboración del cuestionario para la validación de la propuesta.
3. Cálculo de concordancia entre los expertos.
4. Desarrollo práctico y explotación de los resultados.

5. Selección del grupo de expertos a entrevistar

En la investigación se considera que un experto se refiere a: Especialistas que son capaces de brindar criterios terminantes sobre el proceso de obtener portabilidad para las soluciones software basadas en bases de datos Data Warehouse que se basan en PostgreSQL.

Por lo anteriormente planteado, se realiza la selección de expertos teniendo en cuenta distintos aspectos que el autor considera importante. Uno de estos aspectos es la competencia de cada uno de los encuestados, este permite visualizar en una escala de 0 – 10 el nivel de conocimientos que considera que posee cada uno de los encuestados de acuerdo a los aspectos relacionados con el tema central en cuestión: adaptabilidad, instalabilidad y coexistencia de las soluciones software basadas en bases de datos Data Warehouse. Mientras más se acerque a 10, el nivel de conocimientos es más avanzado. Para la selección de los expertos se tienen en cuenta además otros factores como:

- Personas conocedoras, con reconocida competencia y con experiencia en el tema que garantice la confiabilidad de los resultados. Por lo que se considera que debe estar vinculado a la investigación y al desarrollo de bases de datos desarrolladas en PostgreSQL.
- Creativos e interesados en participar.
- Ser graduado de nivel superior y contar además con un año de experiencia como mínimo en el tema.
- Contar con conocimientos y habilidades en actividades de calidad del software.

Se selecciona para la evaluación, profesores que imparten asignaturas de Base de Datos con conocimientos sobre adaptabilidad, instalabilidad y coexistencia, personal de la UCI dedicado a la producción de soluciones software que se basan en Data Warehouse. Elegir los especialistas con las características mencionadas propicia tener resultados gratificantes y de mayor calidad acerca de la solución propuesta al problema presentado.

El grupo de especialistas seleccionados funciona como un todo ya que sus conocimientos combinados formarían el especialista ideal para ejercer en cualquier

esfera del tema central de este trabajo de diploma. De esta manera se pretende obtener distintos puntos de vistas en dependencia de la esfera en que se desempeñe cada uno de los especialistas para poder llegar a conclusiones favorables que permitan asegurar la guía.

La cantidad de expertos seleccionada fue 7, dentro de los que se encontraban 3 desarrolladores de Data Warehouse, 3 pertenecientes al centro DATEC y por último un profesor experto en PostgreSQL de la universidad.

3.2 Elaboración del Cuestionario para la Validación de la Propuesta

En la validación de la guía de buenas prácticas para obtener portabilidad en las soluciones software basadas en bases de datos Data Warehouse, se utilizó la encuesta que se observa en el anexo 3. Para la elaboración de la misma se tuvo en cuenta la teoría de la comunicación, con mecanismos que reduzcan ambigüedades en las respuestas, preguntas claras, precisas e independientes. La encuesta consta de 11 preguntas de enfoque investigativo sobre la validez de la solución propuesta al problema planteado, y ofrecen a los encuestados la oportunidad de hacer una valoración crítica del tema. Estas preguntas tienen como ventajas para los especialistas proporcionar una mayor riqueza a las respuestas ofrecidas. El cuestionario les brinda a los expertos la posibilidad de expresar sus opiniones sobre la guía de manera libre para conocer sus criterios relativos sobre los diferentes temas tratados. A los especialistas se les puso a su disposición la documentación de la guía y se les requirió un tiempo determinado para las respuestas o hacer las preguntas pertinentes que les hubiesen surgido al presentar el documento. Para conocer las preguntas en detalles que componen la encuesta vea el anexo 3.

3.2.1 Resultados de la evaluación a través del método Delphi

Los resultados de la validación se observan a través de las respuestas de cada especialista en las diferentes preguntas que se realizaron.

Pregunta 1: Relación correcta entre los factores determinados y su ubicación dentro de las fases del ciclo de vida del software.

La guía está encaminada a obtener la adaptabilidad, instalabilidad y la coexistencia de la soluciones software de apoyo a las toma de decisiones, brindando una serie de factores y buenas prácticas a tener en cuenta durante el ciclo de vida del software. En este aspecto de la encuesta realizada, los expertos determinarán si los factores están correctamente ubicados. El gráfico nos muestra que cuatro especialistas la califican de muy adecuada (5), tres de ellos como bastante adecuada (4). A continuación la figura muestra los resultados arrojados por la encuesta.

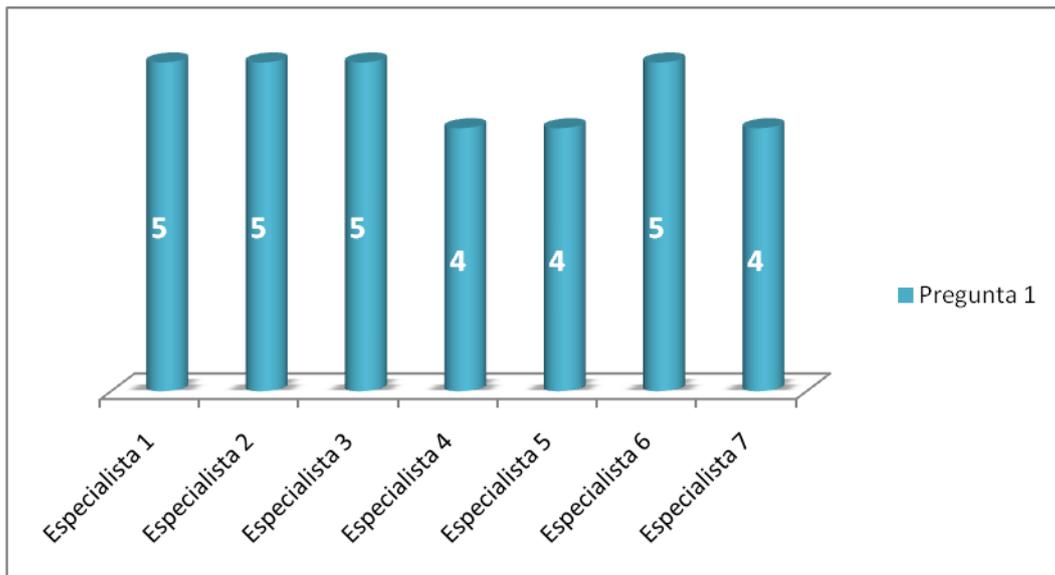


Figura 6: Ubicación correcta de los factores dentro de las fases del ciclo de vida del desarrollo del software.

Pregunta 2: Efectividad de la guía

Los expertos deben determinar el nivel de efectividad de la puesta en práctica de cada uno de los factores, para determinar el grado de efectividad de la guía. En este caso, la respuesta era de [Si (5)-No (4)]. A continuación la figura muestra los resultados arrojados por la encuesta.

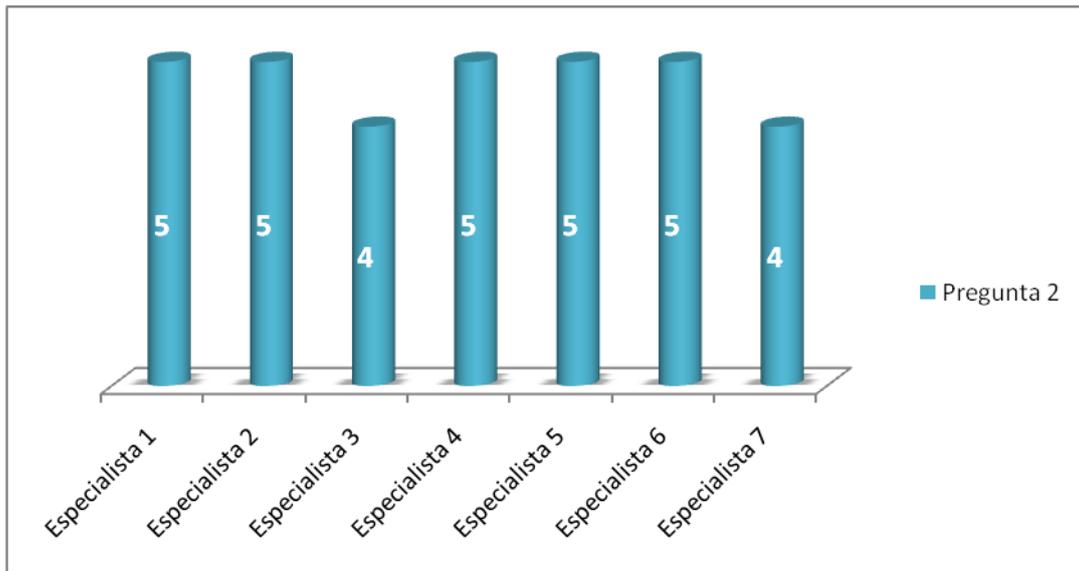


Figura 7: Efectividad de la guía.

Pregunta 3: Necesidad de contar con una guía.

En diferentes proyectos la portabilidad de las soluciones software constituye un aspecto al cual no se le dedica toda la atención que requiere a diferencia de otros procesos en el desarrollo del software. Contar con una guía que permita ayudar a los desarrolladores en este sentido es un gran paso de avance. Por lo antes mencionado el resultado arroja que solo un especialista cree que es **necesaria (4)** la guía y demás opinan **muy necesaria (5)**. A continuación la figura muestra los resultados arrojados por la encuesta.

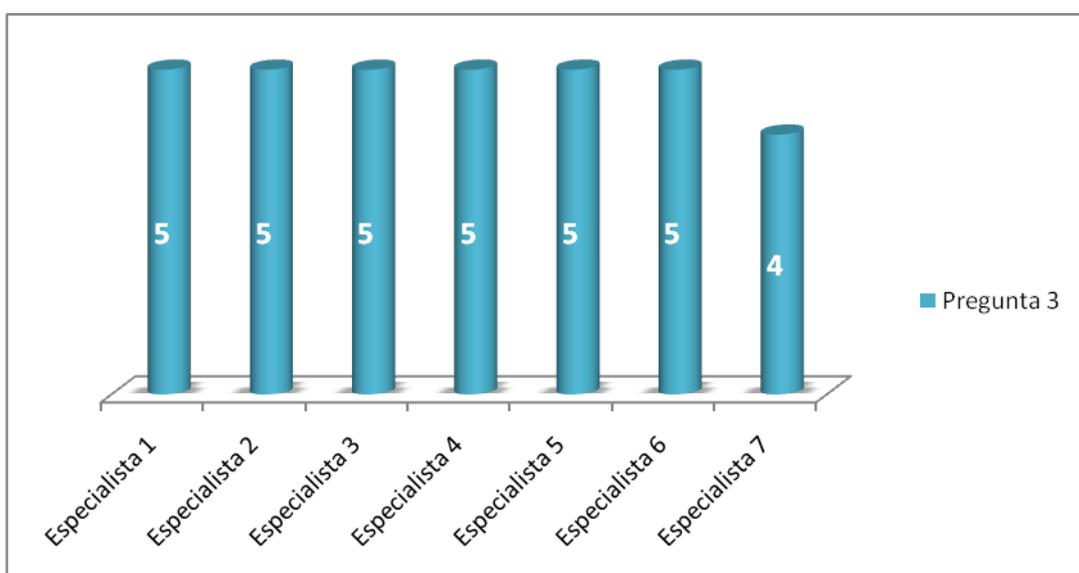


Figura 8: Necesidad de contar con una guía.

Pregunta 4: Nivel de complejidad de la guía.

En este aspecto los especialistas determinan si las buenas prácticas propuestas por la guía son muy difíciles de realizar, o de entender. En este caso, la respuesta era de [Si (5)-No (4)]. A continuación la figura muestra los resultados arrojados por la encuesta.

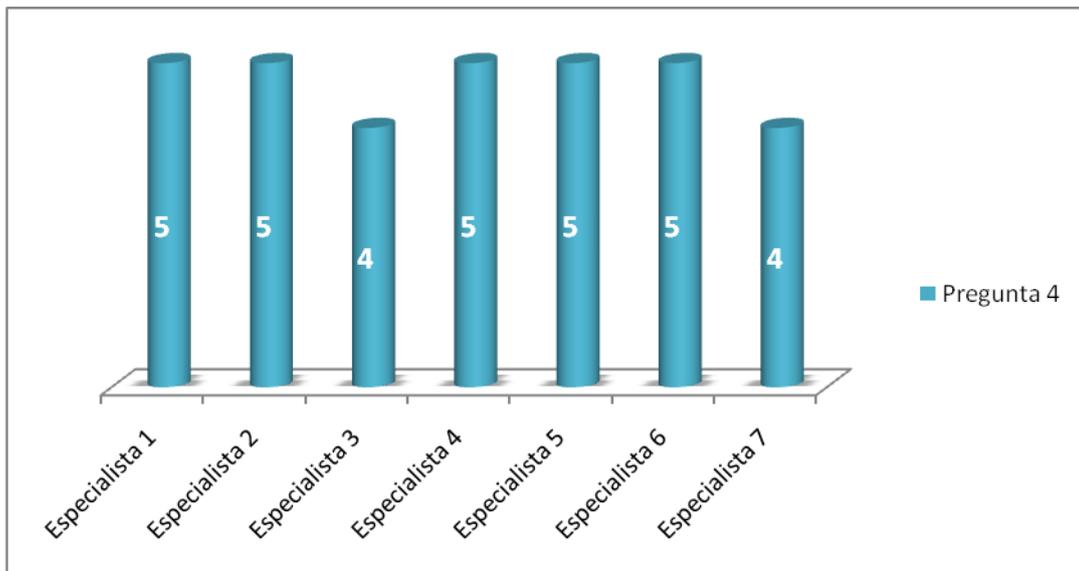


Figura 9: Nivel de complejidad de la guía

Pregunta 5: Nivel de satisfacción de la guía

La guía muestra las buenas prácticas para garantizar la portabilidad durante el proceso de desarrollo de software, lo que garantiza que el nivel de satisfacción sea elevado ya que responde a las necesidades del cliente. Uno de los especialistas opinan como **muy satisfactoria (5)** y el resto de **bastante satisfactoria (4)**. A continuación la figura muestra los resultados arrojados por la encuesta.

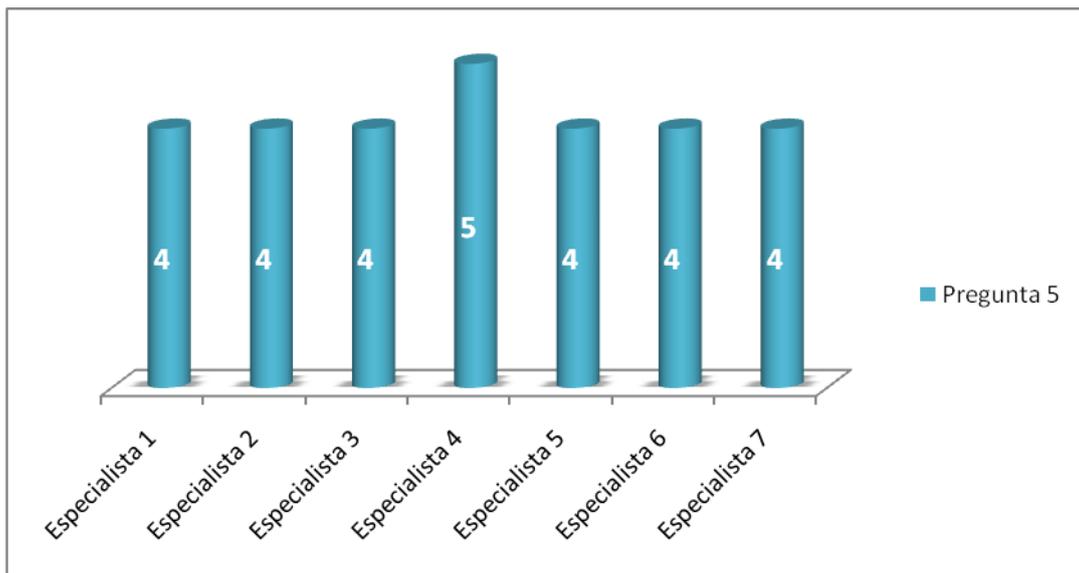


Figura 10: Nivel de satisfacción de la guía

Pregunta 6.a: Repercusión de la guía

Según los especialistas una vez puesta en práctica la solución, esta tendrá una gran repercusión, pues se plantea que no existe tal guía que proponga actividades a realizar en el proceso de desarrollo con el fin de asegurar la portabilidad de las soluciones software. Los especialistas consideran que la guía tiene **elevada repercusión**. En este caso, la respuesta era de seleccionar (5) o no (4). A continuación la figura muestra los resultados arrojados por la encuesta.

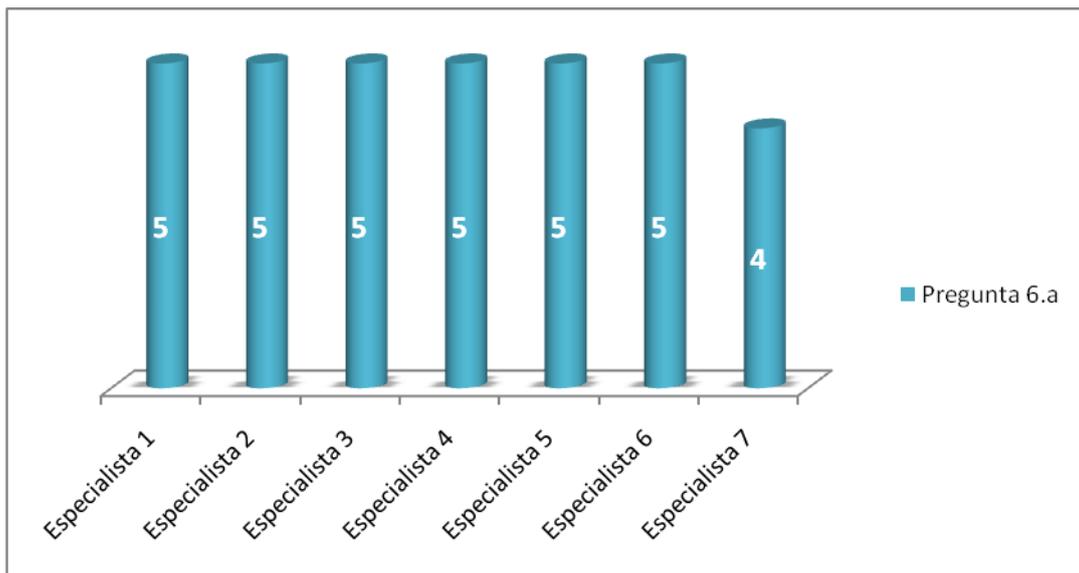


Figura 11: Repercusión de la guía

Pregunta 6.b: Posibilidad de adaptabilidad de la guía.

Los especialistas seleccionan la categoría de adaptable en los proyectos productivos de la universidad que decidan trabajar con data warehouse para las soluciones software de toma de decisiones. Los especialistas marcan la categoría de adaptabilidad a proyectos productivos. En este caso, la respuesta era de seleccionar (5) o no (4). A continuación la figura muestra los resultados arrojados por la encuesta.

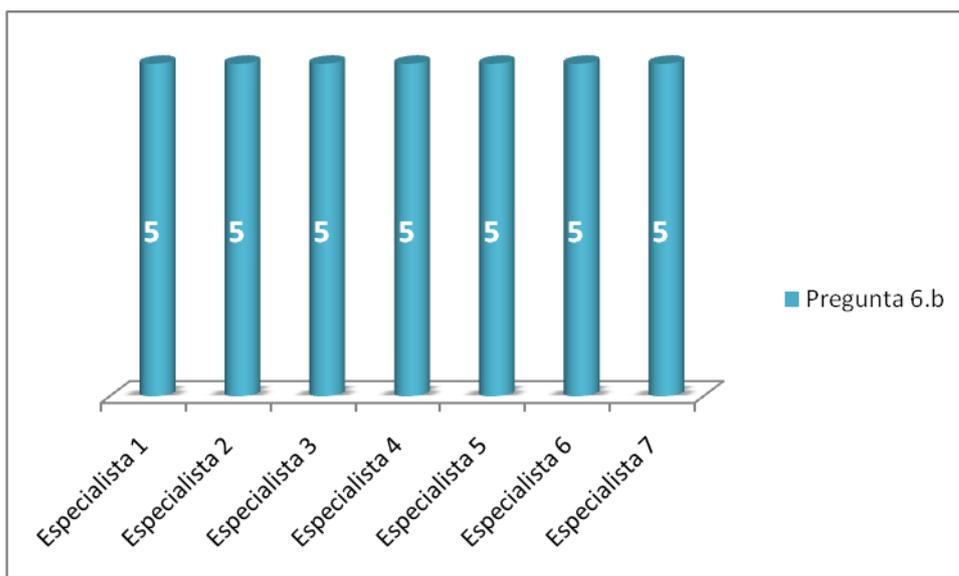


Figura 12: Adaptabilidad de la guía a diferentes proyectos.

Pregunta 7: Nivel de cumplimiento de las actividades en las fases de la guía.

Se percibe en el gráfico que los especialistas califican como muy elevada la portabilidad que se obtendrá si se le otorga un cumplimiento adecuado a las actividades que presenta la guía. Por tanto se obtiene como resultado de las opiniones de los especialistas cinco de ellos proponen **que si (5)**, **uno en cierta medida (4)** y otro que **no (3)**. A continuación la figura muestra los resultados arrojados por la encuesta.

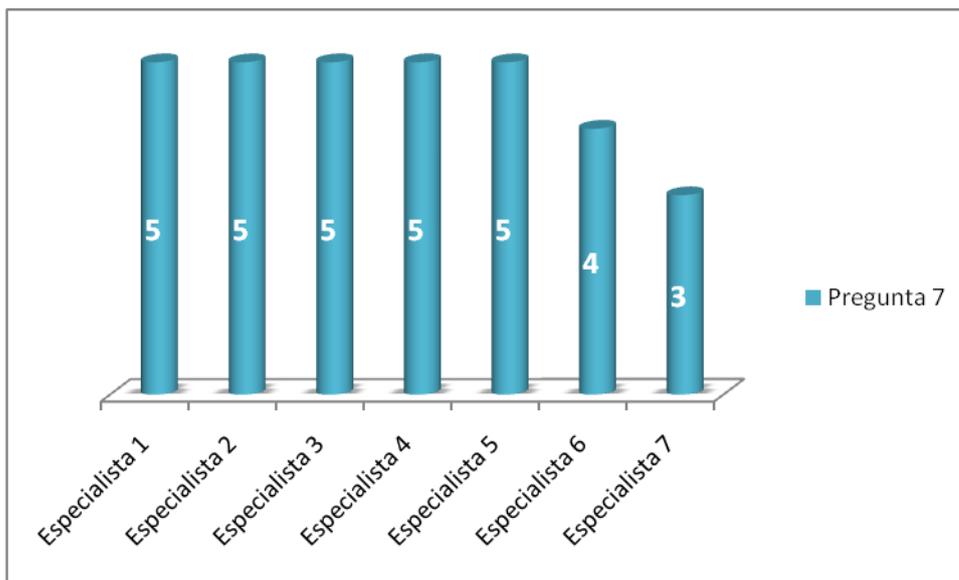


Figura 13: Nivel de cumplimiento de las actividades en las fases de la guía.

Por ciento de aceptación de la guía por pregunta teniendo en cuenta las respuestas.

Al observar la siguiente gráfica se puede determinar claramente el elevado por ciento de aceptación que presenta la guía para los especialistas; demostrando un alto porcentaje de certeza en cada pregunta de la encuesta realizada, lo que demuestra que la guía cumple con el propósito por la que fue creada y da respuestas a los objetivos específicos de la investigación científica. De manera general, la guía desarrollada posee un 89.7 de aceptación por parte de los especialistas seleccionados, por lo que se puede definir que la guía es altamente aplicable y adaptable ya que satisface en un gran por ciento a los desarrolladores.

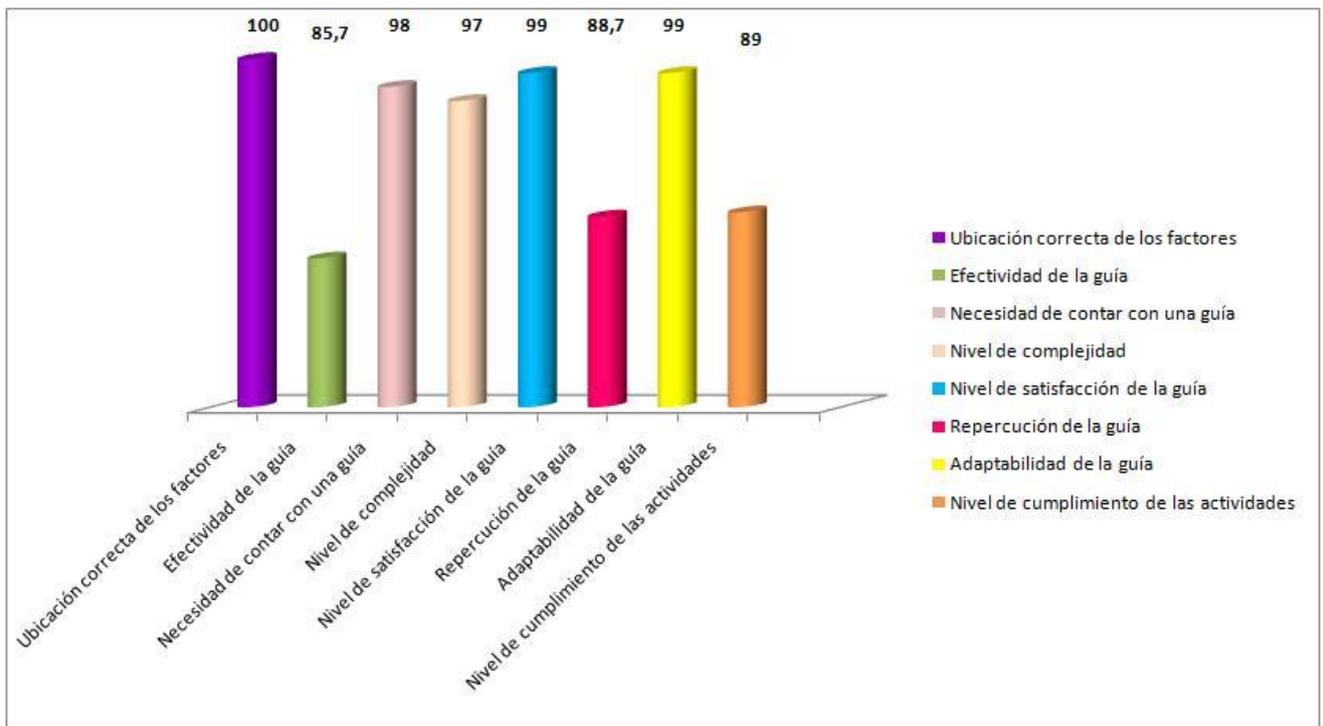


Figura 14: Por ciento de aceptación de la guía por pregunta teniendo en cuenta las respuestas.

Coefficiente de Kendall.

A través de la herramienta SPSS, versión 13.0, se calcula el coeficiente de Kendall que determina el grado de concordancia en las respuestas dadas por los especialistas. Se define que mientras más cerca se encuentre de 1.00 (uno) entonces existe un alto grado de concordancia entre las respuestas de los especialistas sobre la guía. Como resultado se obtuvo el coeficiente de **Kendall con un 0.877**, (Fig. 16); lo que determina que hubo un alto grado de concordancia por parte de los especialistas a la hora de opinar que la guía es factible y responde al problema por la que fue creada.

N			7
Kendall's W ^a			,877
Chi-Square			61,400
df			10
Asymp. Sig.			,000
Monte Carlo Sig.	Sig.		,000 ^b
	95% Confidence Interval	Lower Bound	,000
		Upper Bound	,000

a. Kendall's Coefficient of Concordance

b. Based on 10000 sampled tables with starting seed 2000000.

Figura 15: Coeficiente de Kendall

3.3 Conclusiones

Después de haber analizado los resultados del método Delphi, en el transcurso del presente capítulo se puede afirmar que se realizó la evaluación práctica de la guía, por lo que toda la información ha sido preparada para proseguir con la toma de decisiones. También se llevó a cabo la validación de la guía para elevar la portabilidad de las soluciones software.

En general, el proceso propuesto fue calificado por los expertos como un procedimiento muy adecuado que tiene en cuenta aspectos estudiados en importantes fuentes bibliográficas y que proporciona un cierto orden a todo lo que se había visto en las mismas. Un alto por ciento de los objetivos propuestos fue calificado de muy adecuado, así como las buenas prácticas propuestas y los ejemplos propuestos. Las recomendaciones propuestas estuvieron alrededor de mejorar las buenas prácticas ofrecidas.

Conclusiones generales

En el presente trabajo de diploma para el cumplimiento de los objetivos, en concordancia con las necesidades de la investigación y con el estudio de obtener la adaptabilidad, instalabilidad y coexistencia de las soluciones software, se llevaron a cabo varias acciones:

- Se estudió las definiciones de calidad, los modelos de calidad, la BD Data Warehouse, así como el GSBD PostgreSQL y las características principales de los DSS.
- Se investigó cuáles son los factores que favorecen la obtención de las subcaracterísticas de calidad durante el proceso de soluciones software, adquiriéndose mayor conocimiento, lo que permitió obtener una solución eficiente al problema de la investigación.
- Se realizó la propuesta de solución estableciendo para la misma toda su estructura, conformada por las fases del ciclo de vida de las soluciones software, sus actividades y los objetivos, de esta forma brindar buenas prácticas con el fin de obtener la adaptabilidad, instalabilidad y coexistencia de las soluciones software y así elevar la portabilidad.
- Se estableció el proceso de validación de la solución propuesta mediante el Método de expertos, para determinar que la guía es factible, por tanto permite obtener la adaptabilidad, instalabilidad y coexistencia de las soluciones software.

Recomendaciones

Finalizado el trabajo de diploma donde se crea la guía para obtener adaptabilidad, instalabilidad y coexistencia de las soluciones software se recomienda:

- Utilizar la guía propuesta en proyectos productivos de la UCI para lograr una elevada portabilidad en las soluciones software.
- Profundizar la investigación y posterior perfeccionamiento de esta guía
- Adaptar la solución propuesta a otro tipo de soluciones de software.
- Ampliar los estudios para intensificar el uso de la guía, identificando otros factores que favorezcan el logro de las características de calidad durante el proceso de desarrollo del software.

Referencias bibliográficas

- [1] **Aguilar, Vicente; Suau, Pablo. MySQL vs. PostgreSQL.** Disponible
Ángel, Miguel. Funciones de los Sistemas Gestores de Bases de Datos.
Disponible en: <http://cnx.org/content/m17543/latest/>.
- [3] **Asociación Peruana de Software Libre.** Introducción a PostgreSQL.
<http://apesol.org.pe>.
- [4] **Date, C. J. Introducción a los Sistemas de Bases de Datos.** Ciudad de La Habana: Félix Varela, 2003.
- [5] **Grocock, Jhon M. La cadena de la Calidad.** 2008.
- [6] **Guzmán, Ignacio García Rodríguez.** Bases de datos modelo en red general. Universidad de Castilla la Mancha.
- [7] **Hansen, Gary W. y Hansen, James W.** Diseño y Administración de Bases de datos. 2da. s.l: Prentice Hall. <http://bibliodoc.uci.cu/pdf/reg00071.pdf>.
- [9] **Inmon. Sistema de soporte a la decisión y sistemas inteligentes.**
<http://www.netmedia.info/articulo-32-3680-0-10796.html>.
- [10] **Mato Gracia, Rosa M.** Diseño de Bases de Datos, Octubre 2005.
- [11] **Microsoft Corporation, Microsoft Access.** Disponible en:
<http://office.microsoft.com>
- [12] **Norma Cubana, ISO/IEC 9126-1:2001,** Ingeniería de Software, Calidad del producto. Oficina Nacional de Normalización. 2005.
- [13] **Norma ISO 8402.** Ver.megaret.net.mx. ISO 8402, Términos Generales. 2008.
Disponible en: <http://ver.megared.net.mx/~jccz/iso8402.html>
- [14] **Pressman, Roger S.** Ingeniería de Software. Un enfoque práctico. 1998.
- [15] **Rubio, G.B.** Calidad en Ingeniería de Software. 2002.
- [16] **Ruiz, Dr. Francisco.** El modelo de datos jerárquico. Universidad de Castilla La Mancha.
- [17] **Torregrosa Sánchez Rafael.** Concepto de Calidad y generalidades. 2002.
- [18] **Verificación y Validación.**
<http://wwdi.ujaen.es/asignaturas/computacionestadistica/pdfs/tema6.pdf>.
- [19] **Vidal LV, Monteagudo MV. Estudio Teórico-Conceptual sobre Data Warehouse** [Pregrado]. Ciudad de La Habana: Universidad de La Habana; 2000.
- [20] **Yadier Alarcón Castell-Florit, Dayron Reyes García, Sistema de Traducción Automática para el SIGB Koha,** Ciudad de La Habana: Universidad de Ciencias Informáticas, 2009.

- [21] **Yaniris Roig Morejón, Roberto Lago, Análisis y diseño del módulo gestión de tesis**, Ciudad de La Habana: Universidad de Ciencias Informáticas ,1.06.2007.
- [22] **Yoslaine de la Caridad Ramírez Morales, Suammy Fernández Díaz, Propuesta de Métricas para medir Calidad en Portales WAP**, Ciudad de La Habana: Universidad de Ciencias Informáticas, junio del 2009.
- [23] **Quesada, Juan Antonio López, ingeniería de software. dis.um.es** [En línea] <http://dis.um.es>.
- [24] [En línea]
http://www.redisybod.unam.mx/seminarios_pdf/BuenasPracticasIngSoft.pdf.
- [25] **MANSO, E. 2005. Calidad del software.** 2005.
- [26] <http://www.webprogramacion.com/44/sistemas-operativos/exclusion-mutua.aspx>.
- [27] <http://www.mitecnologico.com/Main/ExclusionMutuaSeccionesCriticas>.
- [28] <http://www.eveliux.com/mx/topologias-de-red.php>.
- [29] **Lidier González Ballester, Diseño de una base de datos para el control de los RRHH en los polos productivos de la facultad 9**, Ciudad de La Habana: Universidad de Ciencias Informáticas, Mayo 2009.
- [30] **Raúl Mató Rodríguez, Diseño de Bases de Datos para la Empresa de Gas Manufacturado**, Ciudad de La Habana: Universidad de Ciencias Informáticas, Mayo 2009.
- [31] [En línea] <http://www.punchador.com/index.php>.

Bibliografía

- 1-Alma, **Enríquez Toledo.** **2006.** MySQL.
<http://www.uaem.mx/posgrado/mcruz/cursos/miic/MySQL.pdf>.
- 2-**Almaguer, Yaquelin Cintra and Núñez, Yandy León. 2008.** Aplicación de Técnicas de Base de Datos para el soporte de indicadores dinámicos en el Sistema Automatizado para el Control de Gestión de Indicadores de Refinación (SACGIR). Ciudad de la Habana: s.n., 2008.
- 3-**Artículo escrito el 16 de diciembre de 2004 por Jean-François Pillou** , se puede encontrar en : <http://es.kioskea.net/contents/qualite/qualite-introduction.php3>
- 4-**Álvarez, Sara. 2007.** Equipo DesarrolloWeb.com. <http://www.desarrolloweb.com>.
- 5-**Álvarez, Brenda Topsy. 2008.** <http://bren-tipsy.blogspot.com/2008/10/2-departamental-metodologa-de.html>.
- 6- **Benitez, R and Roa, I. 2002.** Bases de Datos de Orientadas a Objetos. 2002.
- 7-**Calcines, Anabel Vega and Ruiz, Alfredo Rodriguez. 2008.** LIMS de Calidad del Centro de Ingeniería Genética y Biotecnología: Desarrollo de la Base de Datos del Modulo Liberación Analítica. Ciudad de la Habana : s.n., 2008.
- 8- **Daft, Richard L.** Teoría y diseño organizacional. s.l. : Thomson Learning Ibero, 2005. ISBN 970686363X.
- 9- **Desarrolloweb.com. 2002.** Modelos de bases de datos.
<http://www.desarrolloweb.com>.
- 10-**Fábregas, Yuniesky and Fernández, Daniel. 2007.** Administración, configuración y optimización de un Sistema de BD Descentralizado en Oracle Database 10g release 2. Ciudad de la Habana : s.n., 2007.
- 11-**Juan Manuel Cueva Lovelle,** octubre 19,1999.
http://gidis.ing.unlpam.edu.ar/downloads/pdfs/Calidad_software.PDF
- 12- **Keenan, Peter B.** Spatial Decision Support Systems. [aut. libro] Manuel Mora, Guisseppi A. Forgionne y JatinderN. D. Gupta. Decision Making Support Systems:Achievements, Trends and Challenges for the New Decade. London : Idea Group Publishing, 2003, págs. 28-39.
- 13-**Leyva, Arnoldo Domínguez. 2008.** Modelo Lógico y Físico de la Base de Datos correspondiente a los Módulos de Investigación Criminalística y Estadística del Proyecto CICPC. Ciudad de la Habana: s.n, 2008.

- 14-**Márquez, Merche. 2002. Base de datos Orientado a Objetos.**
<http://www3.uji.es/~mmarques/e16/teoria/cap2.pdf>.
- 15- **Martínez Prieto, Diego, Olalla Piñeiro García, Sergio Rodríguez Pérez, Yobana.** Sistemas de apoyo a la toma de decisiones.
- 16-**N.K.Malhotra. 2003. Investigaciones de Mercado.** Un enfoque practico. Cuarta edición. 2003.
- 17-**Pérez, Erich Mario Gómez and Gálvez, Ariel Torres. 2008.** Administración y optimización de un Sistema de Base de Datos Descentralizado, en PostgreSQL. Ciudad de la Habana : s.n., 2008.
- 18-**Power, Dan. Types of Decision Support Systems (DSS).**
<http://www.gdrc.org/decision/dss-types.html>.
- 19-**Power, D. J. Decision support systems: concepts and resources for managers.** s.l Quorum Books, 2002.
- 20- **Power, D. J. Support Systems: A Multidimensional Approach.** [aut. libro] Manuel Mora, Guisseppi A. Forgionne y JatinderN. D. Gupta. Decision Making Support Systems: Achievements, Trends and Challenges for the New Decade. London : Idea Group Publishing, 2003, págs. 20-27.
- 21-**Silvente, Serguéi Frómeta. 2008.** Modelo lógico y físico de la base de datos del módulo de Investigaciones Forenses del proyecto CICPC. Ciudad de la Habana : s.n., 2008.
- 22-**Sinnexus. Sistemas de Soporte a la Decisión (DSS).** Sinnexus.
http://www.sinnexus.com/business_intelligence/sistemas_soporte_decisiones.aspx.
- 23- **The comp.groupware (FAQ) . Communications-Driven DSS.**
<http://www.dssresources.com/dsstypes/cdss.html>.
- 24- **Turban, E. y Aronson, J. Decision Support Systems and Intelligent Systems.** 6a Edición. s.l. : Prentice Hall, 2001.
- 25-**W.Hansen, Gary and Hansen, James V. 2006.** Diseño y Administración de Base de Datos. 2da Edición. 2006.
- 26-**Zurro, A. Martin and C, J. 2005.** Atencion Primaria. Conceptos, organización y práctica clínica. Quinta edición. 2005.

27-<http://cnx.org/content/m17461/latest/>

28- http://www.elguille.info/colabora/NET2005/Percynet_ConstruyendoSoftCalidad.htm

29-<http://www.mitecnologico.com/Main/LaNormalsolec9126>

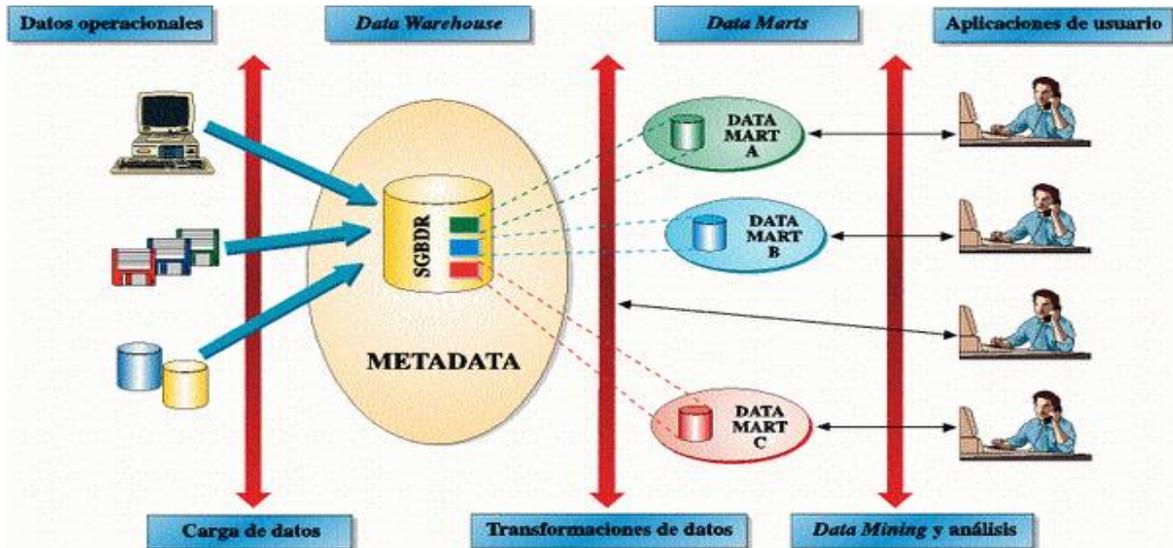
30-<http://www.monografias.com/trabajos5/sistab/sistab.shtml#ins>

31-<http://www.monografias.com/trabajos32/toma-decision/toma-decision.shtml>

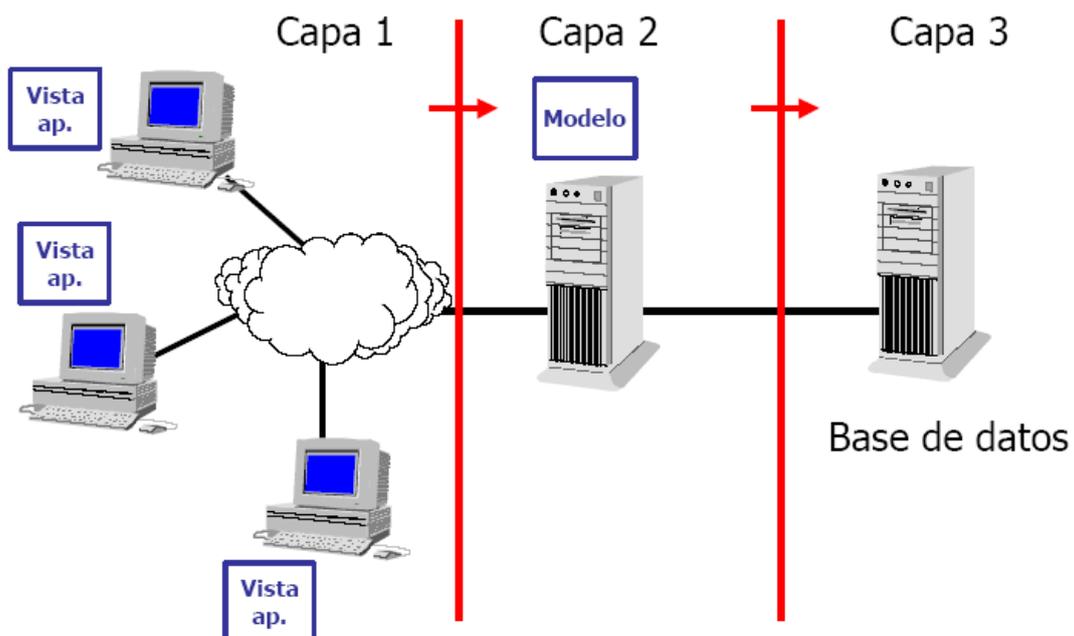
32-http://www.ub.edu.ar/revistas_digitales/UBit/Revista2-3.htm

Anexos

Anexo 1: Ejemplo de una base de datos Data Warehouse



Anexo 2: Arquitectura en capas



Anexo 3: Encuesta aplicada a diferentes especialistas

1-¿Cree usted que los factores potenciales se encuentren ubicados en su correspondiente fase del ciclo de vida de las soluciones software?

Muy adecuada (5) Bastante adecuada (4) Adecuada (3) Poco adecuada (2)
Inadecuada (1).

2- ¿Cree usted que los factores potenciales de las soluciones software logren un grado de efectividad a partir de las buenas prácticas establecidas para el proceso propuesto?

Si No

¿Por qué?

3- Mencione usted si existen razones o cuestiones en el trabajo de los desarrolladores de las soluciones software por las cuales el uso de la guía de portabilidad:

Se debe recomendar No se debe recomendar Se recomienda una parte

¿Por qué?

4- Ve usted dificultades o alguna complejidad en el proceso de aplicación de la guía para el desarrollador de la soluciones software.

Si No

¿Por qué?

5- Cree usted que es abarcadora la guía de buenas prácticas. En una escala del 1 al 5 indique que tan abarcadora es. En caso de su respuesta sea diferente de 5, diga las razones, por las que la guía está incompleta.

1	2	3	4	5

Nota: Tenga en cuenta que 1 quiere decir que la guía no es abarcadora y 5 que lo abarca todo.

6- Diga si la guía de buenas prácticas está facultada para cumplir las siguientes expectativas, para ello marque con una x las que usted cree que pueda cumplir. Escriba el por qué de su selección:

a) ___ Repercusión a los proyectos que decidan usar la propuesta.

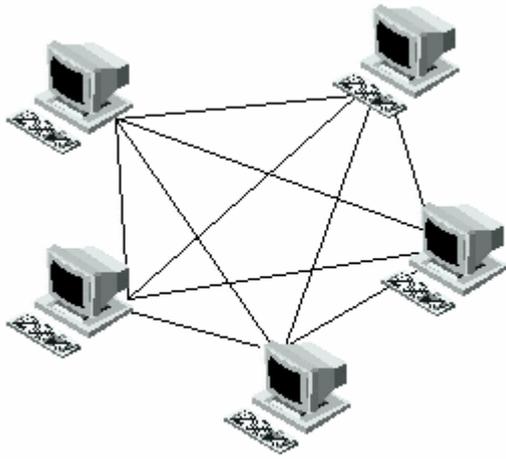
b) ___ Adaptabilidad a proyectos productivos.

7- ¿Cree que usted que mediante el cumplimiento de las actividades asociadas a cada fases del ciclo de vida se pueda lograr elevar la portabilidad de la soluciones software?

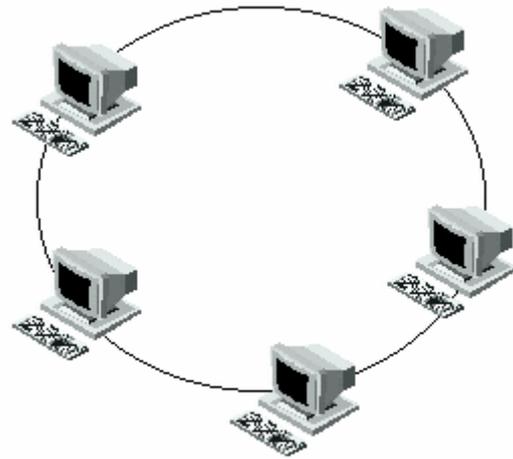
___ Si ___ No ___ En cierta medida

¿Por qué?

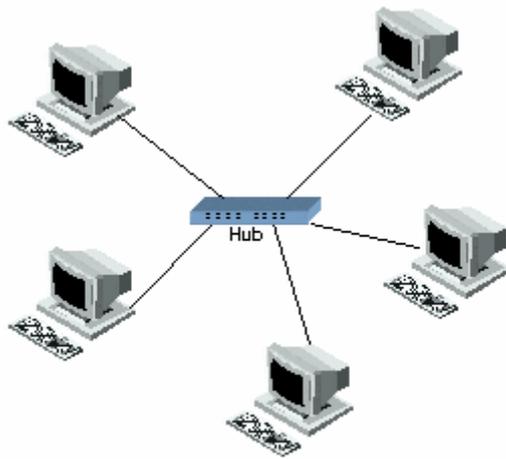
Anexo 4: Diferentes tipos de topología de red



Topología de malla



Topología de anillo



Topología estrella

Glosario de términos

Alcance: Conjunto de actividades interrelacionadas, con un inicio y una finalización definida, que utiliza recursos limitados para lograr un objetivo deseado dentro de un proyecto.

Array: Grupo de servidores proxy que se utiliza para proporcionar caché distribuido entre todos ellos, reparto de carga y tolerancia a fallos. Esto incrementa el tiempo de respuesta a los requerimientos del cliente y la posibilidad de administración unitaria.

Exclusión mutua: Consiste en que un solo proceso excluye temporalmente a todos los demás para usar un recurso compartido de forma que garantice la integridad del sistema.

Patrón: Es una solución a un problema en un contexto, codifica conocimiento específico acumulado por la experiencia en un dominio. Un sistema bien estructurado está lleno de patrones.

Patrones arquitecturales: Aquellos que expresan un esquema organizativo estructural fundamental para sistemas software, denominados estilos.

Patrones de diseño: Aquellos que expresan esquemas para definir estructuras de diseño (o sus relaciones) con las que construir sistemas software.

ISO: International Standard Organization. [Organización Internacional de Estándares]. Fundada en 1947 reúne asociaciones de muchos países y su objetivo es establecer los estándares internacionales, incluidos para la comunicación de datos. Esta organización ha definido los protocolos de comunicación conocidos como ISO/OSI, utilizados por las redes públicas de conmutación de paquetes (X25).

Kimball: Metodología que divide el mundo de Inteligencia de Negocio entre el hecho y las dimensiones. Es muy eficaz y conduce a una solución completa en una cantidad muy pequeña de tiempo.

Protocolo: Conjunto de reglas y normas que determinan cómo se realiza un intercambio de datos, asegurando que los datos recibidos son idénticos a los datos enviados.