



CAPITULO I

EL PROBLEMA

1.1 Planteamiento del problema

Al paso de los años el avance tecnológico en la empresa SERCOMEX se ha visto truncado, por desconocimiento del manejo de nuevas aplicaciones que nos brindan facilidades al momento de gestionar información, razón por la cual no emplea un aplicativo de esta magnitud. Provocando un deterioro de los datos ya que no cuenta con un sistema relacionado a la optimización, control, administración y búsqueda de sus clientes.

La organización se encuentra en desventaja respecto a otras, debido a que no posee ningún sistema georeferenciado que provea al empleado de un aplicativo indispensable en el siglo XXI.

La gestión del sistema informático se debe realizar para alcanzar un propósito uniforme, pues no existe algún proyecto de características similares que esté enfocado a un mejor funcionamiento de la empresa.

El manejo de la información por parte los empleados en la actualidad es complementada simultáneamente con el uso de Internet, por esto el adquirir este sistema se convierte en una necesidad para optimizar procesos.

La Implementación del aplicativo permitirá mejorar el manejo de la información a través de procesos de búsqueda, además podremos innovar el nivel de la organización, logrando que sus empleados obtengan un nuevo conocimiento y desarrollo tecnológico. La implementación de un Sistema georeferenciado de

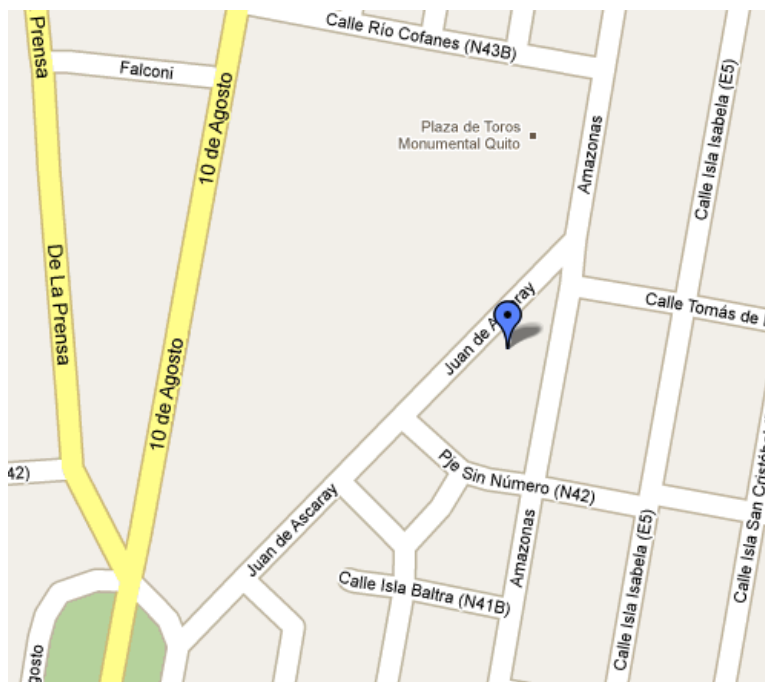
gestión de clientes, permitirá una eficiente administración de los clientes, optimizando tiempo y precisión.

1.2 Formulación del problema

¿Con la implementación del sistema geográfico georeferenciado podremos brindar un mejor servicio a los clientes?

1.3 Delimitación del problema

La implementación del sistema se llevará a cabo en las instalaciones de SERCOMEX. Que tiene lugar en la Provincia de Pichincha, Cantón Quito, calle principal: Juan de Ascaray y Amazonas, con una rápida búsqueda y ubicación.



Empresa SERCOMEX
Ilustración # 1
Fuente: Internet

1.4 Objetivos



1.4.1 Objetivo General

- Desarrollar e implementar el sistema geográfico georeferenciado de gestión de clientes, a fin de optimizar los procesos.

1.4.2 Objetivos Específicos

- Analizar los procesos y reglas del negocio.
- Modelar el diseño de la base de datos.
- Realizar pruebas del aplicativo
- Implementar y capacitar sobre el manejo del software.
- Realizar el código fuente.

1.5 Justificación e importancia

La Implementación del sistema geográfico, permitirá mejorar el conocimiento de los empleados de la Empresa SERCOMEX en cuanto a tecnología e informática, ya que les permitirá obtener nuevos conocimientos sobre el manejo de mapas y sistemas georeferenciados.

La Empresa, con el pasar de los años ha ido creciendo y con esto se han hecho notorios diferentes aspectos tecnológicos, que en el mundo actual se están implementando, tales como manejo de aplicativos que muestran la información de una forma geográfica.

Las autoridades de la empresa brindan todo el apoyo necesario para que proyectos innovadores sean implementados, impulsando el avance tecnológico para la empresa.

Se justifica la realización del presente Proyecto debido a su factibilidad y operatividad enfocadas en una propuesta que permitirá el desarrollo de la empresa ya sea en el ámbito tecnológico y educativo.

Por medio del proyecto la empresa tendrá una herramienta accesible y altamente escalable que elimine el problema del intercambio de datos, visualice información textual, mapas y soporte internacionalización.

Cabe señalar que el hecho de que sea un sistema altamente escalable facilita la adición de nuevos componentes, permitiendo así ofrecer servicios geográficos más completos.

1.5 Alcance

Este sistema va estar compuesto de:

- **Módulo Físico:** El sistema corre en un amplio rango de tipos de hardware, desde servers de computadoras centralizados hasta computadoras desktop utilizadas en configuraciones individuales o de red.

La Empresa requiere de hardware suficientemente específico para cumplir las necesidades de la aplicación. Otras cosas a considerar incluyen: velocidad, costo, soporte, administración, escalabilidad y seguridad.

- **Módulo Lógico:** El sistema provee las funciones y herramientas necesarias para almacenar, analizar y mostrar información geográfica.

Los componentes claves del software son:

1. Un sistema de manejo de base de datos.
 2. Herramientas para el ingreso y manipulación de información.
 3. Soporte para consultas, análisis y visualización geográficos.
 4. Una interface gráfica del usuario (IGU) para fácil acceso.
- **Módulo de Datos:** El componente más importantes del sistema es la información, principalmente se requiere de una buena base de datos, siendo necesario un compromiso a un alto nivel, siendo indispensable para llevar la implementación del sistema a través de esta fase.



- **Módulo de Seguridad:** Seguridad en el manejo de archivo, ya que cada archivo tendrá las personas apropiadas que utilizarán el sistema con correctas copias de seguridad.
- **Módulo de Mantenimiento:** Aclararemos que dentro del uso del sistema solo estarán autorizados el personal de la organización, pero cabe destacar que solo estos serán capacitados explícitamente para dicho fin (uso.) Los realizadores del software serán enteramente responsables del mantenimiento completo del sistema señalado dentro de este proyecto.
- **Módulo de Cartografía:** Dar información geográfica que nos permite tener un conocimiento más o menos exacto de la ubicación exacta. El resultado final de los estudios cartográficos se materializa en una representación sobre el papel que denominamos mapa o plano.
- **Módulo Georeferenciado:** Este módulo interactúa con herramientas de Georeferenciación permitiendo verificar e informar en tiempo real la localización de los clientes, La posición absoluta y así brindar soluciones rápidas a sus problemas

CAPITULO II

MARCO TEORICO

2.1 Antecedentes

SERCOMEX, pertenece al Distrito Metropolitano de Quito, es un Barrio en el norte de la ciudad, al momento no cuenta con el sistema georeferenciado de búsqueda de clientes, para uso de dicha empresa, la organización se compone de jóvenes y de adultos mayores estos últimos son personas que se encuentran dentro de los 40 años de edad, cabe destacar que la directiva no ha realizado la gestión necesaria para contar con un proyecto de esta magnitud, luego de realizar una investigación, no existe el estudio ni la creación de un sistema similar, por consiguiente, como alumno del Instituto Cordillera y complemento para el termino de mis estudios superiores, para optar por el título de Tecnólogo, desarrollaré este proyecto final el cual tendré el apoyo total de la empresa, de sus directivos y por supuesto bajo la supervisión de nuestro tutor quien nos sabrá guiar durante el desarrollo y la creación del proyecto.

2.2 Reseña Histórica

Estamos conscientes de la problemática que aqueja a nuestra sociedad por la inseguridad y sus atentados contra la propiedad y las personas.

Desde 1996 nuestra empresa ha venido sirviendo al país en diferentes áreas.

Lo hemos desarrollado de la mejor manera y nuestra satisfacción es haber trabajado con muchas instituciones a las cuales se les ha servido con el profesionalismo que nos compete.



La tecnología y la sistematización de los procesos han tenido mucho auge y desarrollo en estos últimos años. Desde hace muchos años han revolucionado enormemente las actividades que se realizan diariamente. Los diferentes procesos últimamente se han convertido en una herramienta primordial las hace sentir más seguras y las hace, hasta cierto punto, más productivas.

A pesar de que en la actualidad la tecnología de hoy es capaz de brindar otro tipo de servicios, como datos, audio y video. Sin embargo los procesos que ahora son posibles en aplicaciones que requieran un mayor consumo. La evolución de los procesos ha permitido disminuir el tiempo desde que se los ha ido implementado.

1 MISIÓN

SERCOMEX, proporcionara servicios integrados de seguridad privada, dirigidos a diversos mercados lícitos, con responsabilidad social.

2

3 VISIÓN

Líderes en el manejo de riesgos, con elevados estándares de calidad, empleando capacitación y tecnología de punta, con valores y estructura sólida, para contribuir con el desarrollo del país.

2.3 Marco Referencial

Sistemas de Información Geográfica

El Sistema Corporativo de Información Geográfica define a un SIG como una disciplina basada en conocimientos, metodologías y procedimientos asistidos por computadora que permiten la incorporación, almacenamiento, manipulación, procesamiento, consulta y presentación de información referenciada geográficamente en formatos gráficos y no gráficos.

El Enviromental Systems Research Institute (ESRI) lo define como: una herramienta computacional para trazar y analizar cosas que existan y sucesos que ocurren sobre la tierra.

Algunos argumentos básicos que fundamentan la utilización de un SIG son:

- Permite gestionar un gran volumen de información a diferentes escalas y proyecciones.
- Realiza comparaciones entre escalas y perspectivas emulando una cierta capacidad de representación de diferentes lugares al mismo tiempo.
- Diferencia entre cambios cualitativos y cuantitativos; aportándonos una gran capacidad de cálculo.
- Integra espacialmente datos tabulares y geográficos junto a cálculos sobre variables espaciales o no (topología).
- Admite multiplicidad de aplicaciones y desarrollos; poniendo a nuestra disposición herramientas informáticas estandarizadas.

Los SIG están volviéndose cada vez más una herramienta importante casi para cualquier campo de aplicación debido a que puede asociar información diversa a un espacio geográfico determinado con el que está relacionado.

Funcionamiento de un SIG

El SIG funciona como una base de datos con información geográfica (datos alfanuméricos) que se encuentra asociada por un identificador común a los objetos gráficos de un mapa digital. De esta forma, señalando un objeto se conocen sus atributos e inversamente, preguntando por un registro de la base de datos se puede saber su localización en la cartografía.



La razón fundamental para utilizar un SIG es la gestión de información espacial. El sistema permite separar la información en diferentes capas temáticas y las almacena independientemente, permitiendo trabajar con ellas de manera rápida y sencilla, facilitando al profesional la posibilidad de relacionar la información existente a través de la topología de los objetos, con el fin de generar otra nueva que no podríamos obtener de otra forma.

Las principales cuestiones que puede resolver un Sistema de Información Geográfica, ordenadas de menor a mayor complejidad, son:

- **Localización:** preguntar por las características de un lugar concreto.
- **Condición:** el cumplimiento o no de unas condiciones impuestas al sistema.
- **Tendencia:** comparación entre situaciones temporales o espaciales distintas de alguna característica.
- **Rutas:** cálculo de rutas óptimas entre dos o más puntos.
- **Pautas:** detección de pautas espaciales.
- **Modelos:** generación de modelos a partir de fenómenos o actuaciones simuladas.

Por ser tan versátiles, el campo de aplicación de los Sistemas de Información Geográfica es muy amplio, pudiendo utilizarse en la mayoría de las actividades con un componente espacial. La profunda revolución que han provocado las nuevas tecnologías ha incidido de manera decisiva en su evolución.

Técnicas utilizadas en los Sistemas de Información Geográfica

Las modernas tecnologías SIG trabajan con información digital, para la cual existen varios métodos utilizados en la creación de datos digitales. El método más utilizado es la digitalización, donde a partir de un mapa impreso o con información tomada

en campo se transfiere a un medio digital por el empleo de un programa de Diseño Asistido por Ordenador (DAO o CAD) con capacidades de georeferenciación.

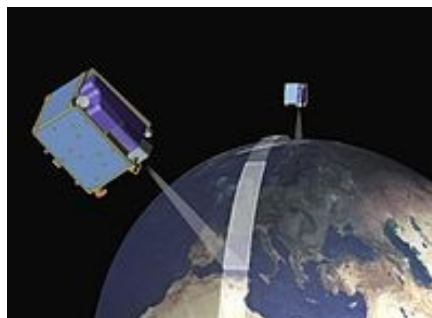
Dada la amplia disponibilidad de imágenes orto-rectificadas (tanto de satélite y como aéreas), la digitalización por esta vía se está convirtiendo en la principal fuente de extracción de datos geográficos.

Esta forma de digitalización implica la búsqueda de datos geográficos directamente en las imágenes aéreas en lugar del método tradicional de la localización de formas geográficas sobre un tablero de digitalización.

4 La representación de los datos

Los datos SIG representan los objetos del mundo real (carreteras, el uso del suelo, altitudes). Los objetos del mundo real se pueden dividir en dos abstracciones: objetos discretos (una casa) y continuos (cantidad de lluvia caída, una elevación). Existen dos formas de almacenar los datos en un SIG: raster y vectorial.

Los SIG que se centran en el manejo de datos en formato vectorial son más populares en el mercado. No obstante, los SIG raster son muy utilizados en estudios que requieran la generación de capas continuas, necesarias en fenómenos no discretos; también en estudios medioambientales donde no se requiere una excesiva precisión espacial (contaminación atmosférica, distribución de temperaturas, localización de especies marinas, análisis geológicos, etc.).





Representación De Datos

Ilustración # 2

Fuente: Internet

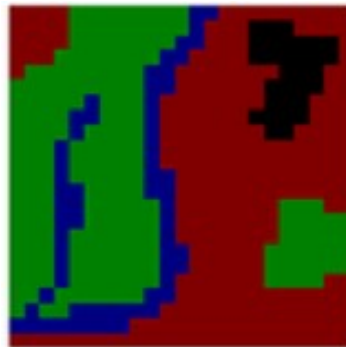
Raster

Un tipo de datos raster es en esencia, cualquier tipo de imagen digital representada en mallas. El modelo de SIG raster o de retícula se centra en las propiedades del espacio, más que en la precisión de la localización. Divide el espacio en celdas regulares donde cada una de ellas representa un único valor.

Cualquiera que esté familiarizado con la fotografía digital reconoce el píxel como la unidad menor de información de una imagen. Una combinación de estos píxeles creará una imagen, a distinción del uso común de gráficos vectoriales escalables que son la base del modelo vectorial. Si bien una imagen digital se refiere a la salida como una representación de la realidad, en una fotografía o el arte transferidos a la computadora, el tipo de datos raster reflejará una abstracción de la realidad. Las fotografías aéreas son una forma comúnmente utilizada de datos raster con un sólo propósito: mostrar una imagen detallada de un mapa base sobre la que se realizarán labores de digitalización. Otros conjuntos de datos raster contendrá información relativa a elevaciones (un Modelo Digital del Terreno), o de reflexión de una particular longitud de onda de la luz (las obtenidas por el satélite LandSat), etc. Los datos raster se compone de filas y columnas de celdas, cada celda almacena un valor único. Los datos raster pueden ser imágenes (imágenes raster), con un valor de color en cada celda (o píxel). Otros valores registrados para cada celda puede ser un valor discreto, como el uso del suelo, valores continuos, como temperaturas, o un valor nulo si no se dispone de datos. Si bien una trama de celdas almacena un valor único, estas pueden ampliarse mediante el uso de las bandas del raster para representar los colores RGB (rojo, verde, azul), o una tabla extendida de atributos con una fila para cada valor único de células. La resolución del conjunto de datos raster es el ancho de la celda en unidades sobre el terreno.

Los datos raster se almacenan en diferentes formatos, desde un archivo estándar basado en la estructura de TIFF, JPEG, etc. a grandes objetos binarios (BLOB), los datos almacenados directamente en Sistema de gestión de base de datos.

El almacenamiento en bases de datos, cuando se indexan, por lo general permiten una rápida recuperación de los datos raster, pero a costa de requerir el almacenamiento de millones registros con un importante tamaño de memoria. En un modelo raster cuantos mayores sean las dimensiones de las celdas menores es la precisión o detalle (resolución) de la representación del espacio geográfico.



Interpretación Cartográfica Raster
Ilustración # 3
Fuente: Internet

Vectorial

En los datos vectoriales, el interés de las representaciones se centra en la precisión de localización de los elementos geográficos sobre el espacio y donde los fenómenos a representar son discretos, es decir, de límites definidos. Cada una de estas geometrías está vinculada a una fila en una base de datos que describe sus atributos. Por ejemplo, una base de datos que describe los lagos puede contener datos sobre la batimetría de estos, la calidad del agua o el nivel de contaminación. Esta información puede ser utilizada para crear un mapa que describa un atributo particular contenido en la base de datos. Los lagos pueden tener un rango de colores en función del nivel de contaminación. Además, las diferentes geometrías



de los elementos también pueden ser comparados. Así, por ejemplo, el SIG puede ser usado para identificar aquellos pozos (geometría de puntos) que están en torno a 2 kilómetros de un lago (geometría de polígonos) y que tienen un alto nivel de contaminación.

Los elementos vectoriales pueden crearse respetando una integridad territorial a través de la aplicación de unas normas topológicas tales como que "los polígonos no deben superponerse". Los datos vectoriales se pueden utilizar para representar variaciones continuas de fenómenos.

Las líneas de contorno y las redes irregulares de triángulos (TIN) se utilizan para representar la altitud u otros valores en continua evolución. Los TIN son registros de valores en un punto localizado, que están conectados por líneas para formar una malla irregular de triángulos. La cara de los triángulos representa, por ejemplo, la superficie del terreno.

Para modelar digitalmente las entidades del mundo real se utilizan tres elementos geométricos: el punto, la línea y el polígono.

I



Interpretación Cartográfica Vectorial
Ilustración # 4
Fuente: Internet

- **Puntos**

Los puntos se utilizan para las entidades geográficas que mejor pueden ser expresadas por un único punto de referencia. En otras palabras: la simple ubicación. Por ejemplo, las ubicaciones de los pozos, picos de elevaciones o puntos de interés. Los puntos transmiten la menor cantidad de información de estos tipos de archivo y no son posibles las mediciones. También se pueden utilizar para representar zonas a una escala pequeña. Por ejemplo, las ciudades en un mapa del mundo estarán representadas por puntos en lugar de polígonos.

- **Líneas o polilíneas**

Las líneas unidimensionales o polilíneas son usadas para rasgos lineales como ríos, caminos, ferrocarriles, rastros, líneas topográficas o curvas de nivel. De igual forma que en las entidades puntuales, en pequeñas escalas pueden ser utilizados para representar polígonos. En los elementos lineales puede medirse la distancia.

- **Polígonos**

Los polígonos bidimensionales se utilizan para representar elementos geográficos que cubren un área particular de la superficie de la tierra. Estas entidades pueden representar lagos, límites de parques naturales, edificios, provincias, o los usos del suelo, por ejemplo. Los polígonos transmiten la mayor cantidad de información en archivos con datos vectoriales y en ellos se pueden medir el perímetro y el área.

Campos de Aplicación de un SIG

Los SIG son útiles en los negocios, el gobierno y en la investigación. Por la parte de los negocios se usan en tareas como estudios de factibilidad para el establecimiento de nuevas sucursales según factores demográficos, vías de acceso, niveles socioeconómicos de la población y existencia de competidores. También se utilizan para redefinir los territorios de ventas, en la búsqueda de menores gastos de transportación, de almacenamiento y en menores tiempos de operaciones, que



resulta en la posibilidad de dedicar mayor atención en la formulación de estrategias para captar clientes y mejorar los productos o servicios.

De igual forma, los SIG son herramientas importantes para la gestión del catastro, el desarrollo urbano, cuantificar y tener información actualizada sobre los recursos naturales de un país, el registro público, instalaciones eléctricas, de agua, gas y comunicaciones, así como el seguimiento de transportes en tiempo real. También se utiliza en la búsqueda de las mejores rutas de distribución. Son útiles para dar respuesta a emergencias, en seguridad pública, en inventarios arqueológicos, en identificación de áreas de riesgo, en reservas ecológicas y recursos naturales; así como en agricultura y nuevos asentamientos. Igualmente se usan para elaborar mapas con fines de mercadotecnia, turismo, difusión, educación y seguramente otras áreas, porque difícilmente habrá tareas de administración, planeación y estudio que no den mejores resultados con la ayuda de un SIG.

Los SIG en Internet

Los servidores de mapas tienen su primer desarrollo en 1993 con el Xerox PARC Map Viewer. Los avances más recientes se han enfocado hacia las estructuras cliente-servidor, SIG's basados en la tecnología orientada a objetos y SIG distribuidos, mejor conocidos como sistemas de IGD's (Información Geográfica Distribuida).

Se denomina Información Geográfica Distribuida (IGD) a los productos o servicios que se relacionan con el empleo de las tecnologías de Internet, las cuales permiten a los usuarios tener acceso a la información geográfica distribuida en una gran variedad de formas como lo pueden ser mapas, imágenes, conjuntos de datos, análisis de información y reportes.

Clasificación para los SIG Distribuidos o IGD's:

- Los más simples son las aplicaciones estáticas, es decir, el servidor web pone a disposición del usuario una imagen en formato GIF o JPG. Estos mapas sólo muestran localizaciones. Ejemplos los encontramos en el sitio de MapQuest.
- Una variación de estas aplicaciones serían aquellas que se actualizan cada determinado tiempo.
- Aplicaciones un poco más complejas son las de los SIG distribuidos, que a partir de peticiones del usuario generan un mapa como respuesta. En estas aplicaciones el usuario genera su propio mapa.
- Otra opción se presenta con el resultado de mapas generados a partir de un procesamiento de datos geográficos, en el que el servidor presenta los datos almacenados como respuesta a una petición del usuario.

El estándar XML

El intercambio de información siempre ha sido un grave problema cuando se utilizan lenguajes y sistemas operativos heterogéneos.

Este problema sólo se agudizó con la llegada de Internet, inclusive previo a su aparición fueron creados mecanismos para lograr el intercambio fluido de información entre diferentes sistemas. El primer método fue GML (General Markup Language), posteriormente SGML (Standard Generalized Markup Language) y actualmente XML. Estos mecanismos son llamados lenguajes de marcado “markup” o meta-lenguajes.

El grupo responsable del desarrollo de éste estándar es el XML WorkingGroup, el cual fue promovido por el W3C (World Wide Web Consortium) en el año de 1996 y



fue presidido por Jon Bosak de SUN Microsystems. La versión 1.0 de XML es una recomendación del W3C desde el 10 de febrero de 1998.

Esta especificación es un subconjunto del estándar SGML que data de 1986 definido por ISO 8879 y desarrollado en un principio por IBM hace tres décadas.

Definición XML

Se define XML de acuerdo a 3 niveles:

- **Primer Nivel:** Un protocolo que nos permite tener acceso al contenido y manejo de la información. La mayoría de los usuarios se limitan al uso de este nivel.
- **Segundo Nivel:** Una familia de tecnologías que pueden hacerlo todo, desde dar formato a un documento hasta hacer filtraciones de datos.
- **Tercer Nivel:** Una filosofía para el manejo de información que busca la máxima utilidad y flexibilidad de los datos, refinándolos a su forma más pura y estructurada.

Cabe señalar que la finalidad de XML no es la de ser un documento final. Se necesitan otros recursos para darle presentación al documento. Tales recursos pueden ser las hojas de estilo (stylesheets), plantillas de traducción XSLT o alguna aplicación en general.

Todo documento XML puede ser representado en forma de árbol. Cada parte del documento ya sea elemento, atributo, texto o comentario se representa como un nodo.

Ventajas de XML

Debido a su gran flexibilidad el estándar XML se puede usar para una infinidad de trabajos y aportar muchas ventajas en varios campos prácticos. Algunas ventajas de XML en algunas aplicaciones son:

- **Migración de datos:** Un caso particular es la migración de datos entre diferentes Bases de datos. Hacerla usando XML sería más sencillo.
- **Presentación de la información diversa:** Con XML podemos generar varias presentaciones de la información de un solo archivo con el uso de hojas de estilo.
- **Filtrado de la información:** Se puede hacer también un filtro de la información de interés dependiendo de cada usuario por medio de plantillas de Traducción XSLT. También por medio de éstas o de alguna otra aplicación se puede generar información en cualquier otro formato a partir de un documento XML. Un ejemplo de esto es la generación de mapas con SVG a partir de un documento GML (Geography Markup Language).
- **Nuevos lenguajes de Marcado:** XML nos da la posibilidad de definir nuestros propios lenguajes para procesar información de una aplicación en específico.

Datos Geográficos y Mapas

Antes de pasar a GML en detalle se debe puntualizar algunos conceptos, el primero es la diferencia entre dato geográfico (que está codificado en GML), y las interpretaciones que se den a ese dato, ya sea mostrándolo en un mapa o cualquier otra forma de visualización. Un dato geográfico es una abstracción del mundo en términos de las propiedades y geometrías que lo caracterizan independientemente de cualquier visualización particular de ese dato. Así como XML está ayudando a la Web a separar claramente el contenido de la presentación, GML hace lo mismo en el mundo de la geografía.



Estructura de GML

GML está basado en un modelo abstracto de geografía desarrollado por el OGC. Este describe el mundo en términos de entidades geográficas llamadas “features”. GML en su segunda versión introduce tres esquemas para validar la estructura de los documentos, estos son geometry, feature y xlink; los dos primeros sustituyen el DTD de GML 1.0. El esquema XLink provee los atributos de xLink para soportar funcionalidades de ligado. Xlink describe una manera estándar de añadir hiperenlaces a un archivo XML.

Es un lenguaje, definido en términos de marcas (tags) XML, que nos permite introducir enlaces en nuestros documentos XML, de modo que podamos relacionar unos documentos con otros. Un XLink permite establecer una relación entre dos o más recursos en la Web, sin que necesariamente estos recursos sepan que están enlazados. Si se desea profundizar más sobre la tecnología XLink se recomienda consultar.

El uso de esquemas XML añade un conjunto rico de tipos de datos primitivos como string, boolean, float y month, entre otros. También permite la creación de tipos de datos definidos por el usuario. XML ofrece varias ventajas al usarse para restringir codificaciones de GML:

- Permite la mezcla entre diferentes vocabularios usando “Namespaces”.
- Permite un mejor control sobre la estructura de la jerarquía de definición de tipos.
- Otorga extensibilidad y flexibilidad por medio de tipos derivados y grupos de substitución.

Features (Rasgos Geográficos)

La OGC define a un feature geográfico como la abstracción de un fenómeno real que está asociado a una ubicación relativa con la Tierra (Spatial Reference System).

Un feature geográfico es aquel cuyas propiedades pueden ser estimadas geoméricamente y el número de propiedades que puede tener un feature está determinado por su definición de tipo (esto se determina en el esquema de aplicación que lo norma).

Un feature geográfico se compone de propiedades descriptivas y geométricas. Los datos descriptivos están compuestos por tuplas del tipo (nombre, tipo, valor). GML define por defecto los datos descriptivos opcionales: descripción, nombre, fid (un tipo ID único para cada feature) y boundedBy atributo de tipo GML Box usado para definir los límites de un feature dado. Los datos geométricos están compuestos por figuras básicas como punto, línea, polígono y multigeometrías. GML permite también definir geometrías propias en caso de ser necesario.

FeatureCollections (Colecciones de Features Geográficos)

GML 2.1.2 soporta la construcción de FeatureCollections. Este elemento como su nombre lo indica es una colección de features, que puede ser considerado así mismo como un feature y puede contener incluso a otras colecciones de features. Cada uno debe usar la propiedad featureMember para cada feature o featurecollection contenido. La especificación obliga al uso del elemento boundedBy y al ser considerado también como un feature puede tener propiedades propias además de las propiedades de los features que contiene.

GML define dos maneras para representar las coordenadas de un elemento geométrico. La primera es por medio de etiquetas <coord> que encapsulan componentes de X,Y,Z y la segunda es por medio de una sola cadena contenida entre etiquetas <coordinates> cada coordenada X,Y,Z va separada por coma y cada tupla se separa por un espacio en blanco o salto del línea.

SQL SERVER 2008

Poder manejar información geográfica, la que hoy en día es de alta importancia en las organizaciones, con todo el tema de globalización



Acceder a la Información desde cualquier lugar en cualquier momento, con SQL 2008 podrá crear rápidamente aplicaciones conectadas a la base de datos con la funcionalidad de funcionar en forma desconectada y después sincronizarlos con la base de datos central sin perder la línea de negocio y manteniendo los datos validados es el principal medio Server. Expone claves para las operaciones realizarse en SQL Server, incluyendo creación y modificación de esquemas de la base de datos, introducir y editar datos en la base de datos, así como datos o administren el servidor, aprovechan la funcionalidad de SQL Server mediante el envío de consultas de T-SQL y declaraciones que son procesadas por el servidor y los resultados (o errores) regresan a la aplicación cliente. SQL Server que sean administrados mediante T-SQL.

Power Designer

Una aproximación hacia el modelado de procesos de negocio que permita alinear el negocio con TI, es una solución empresarial para el diseño de bases de datos y el modelado de datos que le ayudará a implementar una arquitectura empresarial efectiva y a proporcionar un potente modelo de datos conceptual para el ciclo de vida del desarrollo de aplicaciones.

Power Designer es un único conjunto de herramientas que combina distintas técnicas estándar de moldeamiento: De aplicación a través de UML, técnicas de Modelado de Procesos Empresariales y técnicas tradicionales de modelado de base de datos.

La herramienta de modelado de datos, que permite a los usuarios visualizar, analizar y manipular metadatos más fácilmente a fin de lograr una arquitectura de información empresarial efectiva

Microsoft Project

Es un software de administración de proyectos, el cual está diseñado para asistir a administradores de proyectos en el desarrollo de planes, asignación de recursos a tareas, dar seguimiento al progreso, administrar presupuesto y analizar cargas de trabajo.

Microsoft Office Project sirve para la gestión de proyectos, aplicando procedimientos descritos en el PMBOK (Management Body of Knowledge) del PMI (Project Management Institute). Es usado en forma avanzada, para no cometer

errores de programación, aplicando sus bases que están fundamentadas en el método de la ruta crítica.

Visual Studio 2010

Las funcionalidades nuevas y mejoradas de Visual Studio 2010 le ayudan a dar rienda suelta a su creatividad. Personalice su espacio de trabajo con el soporte de varios monitores. Cree aplicaciones ricas con SharePoint y la Web. Elimine el temido problema de no reproducibilidad con IntelliTrace. Desarrolle para un rango amplio de versiones de framework .NET.

Herramienta	Versión	Utilidad
SQL SERVER2008	R2	Motor de Base de Datos para recopilación de información.
Power Designer	12.5	Para realizar el Diseño de la Base de Datos, Modelo Físico, Modelo Conceptual y modelo físico
Microsoft Project	2010	Programa para realizar el cronograma de actividades del proyecto.
VISUAL STUDIO	2010	Programa para desarrollo del Proyecto a realizarse

Tabla Nº1

Título. Herramientas Utilizadas

Fuente. Propia del Autor

2.4 Marco Legal

Superintendencia de Compañías

La institución es un ente económico cuyo esfuerzo se orienta a ofrecer servicios de seguridad que al ser ofrecidos a la comunidad un valor marginal conocido como utilidad. La empresa en cualquier sistema económico será el motor que mueva la economía de un país; la importancia radica en su capacidad generadora de riqueza, que al ser distribuida equitativamente propicia la paz social y por ende tranquilidad y desarrollo. La empresa puede ser clasificada



desde varios puntos de vista; para este efecto se tomará en consideración los siguientes:

Por la Actividad que Cumple:

Comercial: Aquella que se encarga del acercamiento de los bienes desde el productor hacia el intermediario minorista o al consumidor, sin realizar cambios de forma ni de fondo en la naturaleza de los bienes.

Industria: Es aquella encargada de la transformación, modificación substancial o leve de ciertos bienes menores en otros mayores con la ayuda de factores de la producción.

Servicios: Empresa creada con el fin de atender ciertas necesidades de carácter biológico, sentimental, afectivo y similares.

Por el Tamaño:

Las empresas se clasifican en pequeñas, medianas y grandes. El tamaño de una empresa está dado por varios factores, entre los cuales los más destacados son: Valor del Patrimonio, Volumen de sus Activos Fijos, Número de Personas, Superficie o área de utilización, etc.

Por el Sector al que pertenecen: Las empresas pueden pertenecer al:

Sector Privado: El aporte del capital corresponde a personas naturales o jurídicas del sector privado.

Sector Público: Si el aporte del capital lo hace el gobierno (Estado).

Sector Mixto: Cuando a la conformación del capital concurren los aportes tanto del sector privado como del sector público.

Por la Forma de Organización: En este sentido las empresas se clasifican en:

- **Unipersonales:** El capital se conforma con el aporte de una sola persona natural.
- **Sociedad o Compañía:** El capital (propiedad) se conforma mediante el aporte de varias personas naturales o jurídicas. Las sociedades se subdividen:
- **De Personas:** En comandita Simple y Nombre Colectivo.
- **De Capital:** Sociedad Anónima, Economía Mixta, Compañía Limitada y en comandita por Acciones.

OBLIGACIONES LEGALES DE LAS COMPAÑÍAS SUJETAS A CONTROL DE LA SUPERINTENDENCIA

Art 20: “Las compañías constituidas en el Ecuador, sujetas a la vigilancia y fiscalización de la Superintendencia de Compañías, enviarán a ésta en el primer cuatrimestre de cada año”:

- Copias autorizadas del balance general anual, del estado de cuenta de pérdidas y ganancias, así como de las memorias e informes de los administradores, y de los organismos de fiscalización establecidos por la Ley;
- La nómina de los administradores, representantes legales y socios o accionistas; y
- Los demás datos que se contemplaren en el reglamento expedido por el Superintendente de Compañías.

Ley de Ordenanza Municipal

Art. P.2.- ORDENANZAS.- “El Consejo Metropolitano de Quito solo podrá expedir como ordenanzas normas de carácter general que serán, necesariamente, reformatorias de este Código, ya por modificar sus disposiciones, ya por agregarle



otras nuevas, y se denominará ordenanzas metropolitanas. Se excluyen de lo previsto en el inciso anterior las siguientes ordenanzas”:

- a) Las relacionadas con el presupuesto municipal, que se denominarán ordenanzas presupuestarias;
- b) La Ordenanza de Zonificación y sus reformas;
- c) El Código de Arquitectura y sus reformas; las ordenanzas en virtud de las cuales se dé nombre a calles, plazas u otros bienes de dominio público; las ordenanzas que, pese a no contener disposiciones de carácter general, se expiden como tales por mandato de norma expresa; y las ordenanzas que regulan urbanizaciones sujetas a reglamentación particular. Las tres fases de ordenanzas a las que se refiere este artículo tendrán, cada una de ellas, una numeración distinta e independiente.

Art. P.3.- VIGENCIA.- “Las ordenanzas, salvo que en ellas se indique lo contrario, entrarán en vigencia a partir de la fecha de su sanción. Esta norma no es aplicable a las ordenanzas de carácter tributario, que se someterán a las reglas del Código Tributario y la Ley de Régimen Municipal”.

Art. P.4.- PUBLICACIÓN.- “Todas las ordenanzas se publicarán en la Gaceta Municipal y las ordenanzas metropolitanas, demás, en el Registro Oficial”.

Art. P.5.- OBLIGACION DEL SECRETARIO DEL CONCEJO.- “El Secretario del Consejo Metropolitano está obligado a emitir una copia de toda Ordenanza sancionada por el Alcalde al Administrador General, al Procurador Metropolitano, a los directores generales, administradores zonales, gerentes de empresas metropolitanas y comisarios metropolitanos”.

Art. P.6.- ACUERDOS Y RESOLUCIONES.- “Las decisiones del Consejo Metropolitano que no tengan carácter general, se expedirán mediante acuerdos o resoluciones. Se expedirán también como acuerdos o resoluciones las reformas o derogatorias de disposiciones que, pese a no tener carácter general, hayan sido expedidas como ordenanzas municipales, antes de la vigencia de este Código”.

DERECHOS DE PROPIEDAD INTELECTUAL

Para el desarrollo y creación del Sistema georeferenciado, yo ejecutor de este proyecto final, tendré en cuenta, los artículos de la ley de Propiedad Intelectual, proporcionando el derecho y permiso al Instituto Tecnológico Superior Cordillera, para que puedan a futuro dar mejoras al tema de mi proyecto final, con nuevas reformas y demás actividades que competan al tema que estamos desarrollando, para beneficio de las diferentes comunidades que requieran de este tipo de ayudas.

Ley de la Propiedad Intelectual.

En artículo primero el Estado garantiza la propiedad Intelectual, de conformidad con la ley, así:

Art.1. “El Estado reconoce, regula y garantiza la propiedad intelectual adquirida de conformidad con la ley, las Decisiones de la Comisión de la Comunidad Andina y los convenios internacionales vigentes en el Ecuador”.

La propiedad intelectual comprende:

Los derechos de autor y derechos conexos.

En el siguiente artículo contempla la igualdad de derechos para todos, así:

Art. 2. “Los derechos conferidos por esta Ley se aplican por igual a nacionales y extranjeros, domiciliados o no en el Ecuador



En el siguiente artículo nos habla que, el Estado Ecuatoriano, estará bajo la supervisión de un Organismo que velará por los intereses, tanto de la nación como de su gente, así:

Art. 3. “El Instituto Ecuatoriano de la Propiedad Intelectual (IEPI), es el Organismo Administrativo Competente para propiciar, promover, fomentar, prevenir, proteger y defender a nombre del Estado Ecuatoriano, los derechos de propiedad intelectual reconocidos en la presente Ley y en los tratados y convenios internacionales, sin perjuicio de las acciones civiles y penales que sobre esta materia deberán conocerse por la Función Judicial”.

En el Artículo cuarto, se reconoce el derecho del Autor, así:

Art. 4. “Se reconocen y garantizan los derechos de los autores y los derechos de los demás titulares sobre sus obras”.

En el artículo quinto, nos habla de que el desarrollo de una obra, por cualquier autor, no importa su nacionalidad, su destino, su lugar de difusión, o su expresión, no serán alteradas, ni desprotegidas, así:

Art. 5. “El derecho de autor nace y se protege por el solo hecho de la creación de la obra, independientemente de su mérito, destino o modo de expresión”.

Se protegen todas las obras, interpretaciones, ejecuciones, producciones o emisión radiofónica cualquiera que sea el país de origen de la obra, la nacionalidad o el domicilio del autor o titular. Esta protección también se reconoce cualquiera que sea el lugar de publicación o divulgación.

El reconocimiento de los derechos de autor y de los derechos conexos no está sometido a registro, depósito, ni al cumplimiento de formalidad alguna.

El derecho conexo nace de la necesidad de asegurar la protección de los derechos de los artistas, intérpretes o ejecutantes y de los productores de fonogramas.

En el artículo sexto, nos habla de que el autor de una obra, puede ser compatible con otros autores independientes, que puede modificar su obra, siempre que se rija y sea reconocida por la ley, así:

Art. 6. "El derecho de autor es independiente, compatible y acumulable con:

La propiedad y otros derechos que tengan por objeto la cosa material a la que esté incorporada la obra;

Los derechos de propiedad industrial que puedan existir sobre la obra; y,

Los otros derechos de propiedad intelectual reconocidos por la ley.

En el artículo séptimo, nos habla de algunos tipos de términos, que dentro de esta ley, tendrán el siguiente significado y validez para su efecto, es así":

Art. 7. "Para los efectos de este Título los términos señalados a continuación tendrán los siguientes significados":

Autor: Persona natural que realiza la creación intelectual.

Base de datos: Compilación de obras, hechos o datos en forma impresa, en una unidad de almacenamiento de ordenador o de cualquier otra forma.

Compilación: Agrupación en un solo cuerpo científico o literario de las distintas leyes, noticias o materias.



Copia o ejemplar: Soporte material que contiene la obra o producción, incluyendo tanto el que resulta de la fijación original como el que resulta de un acto de reproducción.

Divulgación: El acto de hacer accesible por primera vez la obra al público, con el consentimiento del autor, por cualquier medio o procedimiento conocido o por conocerse.

Obra en colaboración: La creada conjuntamente por dos o más personas naturales.

Obra colectiva: Es la creada por varios autores, por iniciativa y bajo la responsabilidad de una persona natural o jurídica, que la publica o divulga con su propio nombre, y en la que no es posible identificar a los autores o individualizar sus aportes.

El balance general anual y el estado de pérdidas y ganancias estarán aprobados por la junta general de socios o accionistas, según el caso.

LEY DE COMERCIO ELECTRÓNICO

Considerando:

Que el uso de sistemas de información, incluida la Internet ha adquirido importancia para el desarrollo del comercio y la producción, permitiendo la realización y concreción de múltiples negocios de trascendental importancia, tanto para el sector público como para el sector privado.

Que es necesario impulsar el acceso a la información, a los servicios electrónicos que se generan por y a través de diferentes medios electrónicos.

Que se debe generalizar la utilización de servicios geográficos e Internet, de modo que éstos se conviertan en un medio para el desarrollo del comercio, la educación y la cultura.

Que a través de los servicios geográficos, incluida la Internet se establecen relaciones económicas y de comercio, y se realizan actos y contratos de carácter civil y mercantil que es necesario normarlos, regularlos y controlarlos, mediante la expedición de una Ley especializada sobre la materia.

Que es indispensable que el Estado Ecuatoriano cuente con herramientas jurídicas que le permitan el uso de los servicios electrónicos, incluido el comercio electrónico y acceder con mayor facilidad a la cada vez más compleja red de los negocios internacionales; y en uso de sus atribuciones, expide la siguiente:

Art. 1.- Objeto de la Ley.- “Esta Ley regula los mensajes de datos, la firma electrónica, los servicios de certificación, la contratación electrónica y telemática, la prestación de servicios electrónicos, a través de redes de información, incluido el comercio electrónico y la protección a los usuarios de estos sistemas”.

Art. 4.- Propiedad Intelectual.- “Los mensajes de datos estarán sometidos a las leyes, reglamentos y acuerdos internacionales relativos a la propiedad intelectual”.

Art. 5.- Confidencialidad y reserva.- Se establecen los principios de confidencialidad y reserva para los mensajes de datos, cualquiera sea su forma, medio o intención.

Toda violación a estos principios, principalmente aquellas referidas a la intrusión electrónica, transferencia ilegal de mensajes de datos o violación del secreto profesional, será sancionada conforme a lo dispuesto en esta Ley y demás normas que rigen la materia.



Art. 7.- Información original.- “Cuando requiera u obligue que la información sea presentada o conservada en su forma original, este requisito quedará cumplido con un mensaje de datos, si siendo requerido conforme a la Ley, puede comprobarse que ha conservado la integridad de la información, a partir del momento en que se generó por primera vez en su forma definitiva, como mensaje de datos. Se considera que un mensaje de datos permanece íntegro, si se mantiene completo e inalterable su contenido, salvo algún cambio de forma, propio del proceso de comunicación, archivo o presentación.

Por acuerdo de las partes y cumpliendo con todas las obligaciones previstas en esta Ley, se podrán desmaterializar los documentos que por ley deban ser instrumentados físicamente. Los documentos desmaterializados deberán contener las firmas electrónicas correspondientes debidamente certificadas ante una de las entidades autorizadas según lo dispuesto en el artículo 29 de la presente ley, y deberán ser conservados conforme a lo establecido en el artículo siguiente”.

Art. 9.- Protección de datos.- “Para la elaboración, transferencia o utilización de bases de datos, obtenidas directa o indirectamente del uso o transmisión de mensajes de datos, se requerirá el consentimiento expreso del titular de éstos, quien podrá seleccionar la información a compartirse con terceros.

La recopilación y uso de datos personales responderá a los derechos de privacidad, intimidad y confidencialidad garantizados por la Constitución Política de la República y esta ley, los cuales podrán ser utilizados o transferidos únicamente con autorización del titular u orden de autoridad competente.

No será preciso el consentimiento para recopilar datos personales de fuentes accesibles al público, cuando se recojan para el ejercicio de las funciones propias de la administración pública, en el ámbito de su competencia, y cuando se refieran a personas vinculadas por una relación de negocios, laboral, administrativa o

contractual y sean necesarios para el mantenimiento de las relaciones o para el cumplimiento del contrato”.

Art. 12.- Duplicación del mensaje de datos.- “Cada mensaje de datos será considerado diferente. En caso de duda, las partes pedirán la confirmación del nuevo mensaje y tendrán la obligación de verificar técnicamente la autenticidad del mismo”.

CAPITULO III

METODOLOGIA

3.1 TIPOS DE INVESTIGACIÓN

3.1.1 Investigación Cuantitativa

Investigación Bibliográfica Documental

Este tipo de investigación aplique en mi proyecto ya que mediante el mismo me permite realizar un tipo de investigación analítica y sintética de libros, manuales técnicos, documentos escritos los mismos que me permitieron conocer, interpretar, comparar y enfocar criterios y opiniones válidas en el manejo y estructuración de



los datos; realizar todo este proceso lleva consigo el poder materializar los conceptos obtenidos en los documentos anteriormente mencionados y esquematizarlos en un diseño y modelo de datos que será la base esencial de la estructuración del modelo del negocio que debo construir y obtener un sistema de información geográfica eficiente.

Igual tratamiento lo realice con la información que obtuve a través de la internet poniendo mucho énfasis en las conceptualizaciones que me permitió construir conclusiones y recomendaciones válidas en el desarrollo de mi sistema, cabe recalcar que todo esto es con la finalidad de poder orientar de mejor manera el proceso de la teoría con la integración en la práctica a través del desarrollo de sistemas informáticos.

3.1.2 Investigación De Campo

Investigación Descriptiva

La investigación descriptiva aplique a mi proyecto en la utilización de la búsqueda de soluciones a la problemática planteada, iniciando con la estructuración de explicaciones lógicas y definidas al árbol de problemas que tengo que analizar a fin de poder encontrar las causas que originaron los distintos eventos y estos a su vez desencadenaron la problemática general, cabe recalcar que el enfoque general de mi tema presenta condiciones y características reales las mismas que pueden medirse y cuantificarse en cualquier momento, la utilización del análisis y la síntesis en este proyecto y en este tipo de investigación permitió que los procesos y procedimientos enmarcados en el sistema tengan una correcta descripción y sintonización con todo el resto de elementos que tendrán que realizar en el desarrollo del software informático.

Investigación Explicativa

La investigación explicativa apliqué desde el punto de vista del análisis de la problemática que se desarrolla con aristas predominantes en interrogantes que

tenemos que responder previo al desglose analítico y sintético del tema, el mismo que se relaciona con preguntas por qué ocurrió un hecho y en qué condiciones se encuentra el mismo para lo cual siempre será necesario verificar la causa que lo produjo, la circunstancia en el entorno donde se desarrolló y los afectos, entendiéndose estos como los impactos que pueda causar el no tener un sistema como el anteriormente mencionado.

3.2 METODOS DE INVESTIGACION

3.2.1 El Método Histórico Lógico

Este método apliqué a mi proyecto "SISTEMA GEOGRÁFICO GEOREFERENCIADO DE GESTION DE CLIENTES", ya que con la ayuda del mismo me permitió validar todos los procesos y procedimientos que se encuentran materializados en el sistema y que me permitieron reproducir cronológicamente toda la sucesión de tareas que se deben ejecutar en un macro proceso, al mismo tiempo me permito definir claramente el comportamiento de cada uno de los procesos, es decir verificando el alcance que tiene cada uno de ellos en la realización de determinada tarea, y al mismo tiempo definida en un tiempo de ejecución. Por consiguiente este método me ayudó a delimitar las áreas automatizables y al mismo tiempo las actividades que deben cumplir cada uno de los actores involucrados.

3.2.2. Método Analítico Sintético

El método analítico sintético apliqué en mi proyecto, ya me permitió realizar un análisis a profundidad de cada uno de los elementos que intervienen en el desarrollo del sistema, es decir una metodología, diseño, estructuración de datos, etc., lo que llevó a la incorporación de un lenguaje de programación que va a permitir la sistematización en la organización; por otro lado el realizar una síntesis de la información teórica y conceptual me permitió reunir todos los argumentos válidos, a fin de poder estructurar toda la lógica didáctica y conceptual de todo elemento que interviene en el desarrollo de mi tema "SISTEMA GEOGRÁFICO GEOREFERENCIADO DE GESTION DE CLIENTES" pudiendo concentrar el mayor



esfuerzo en la búsqueda de información y análisis de la misma. Por consiguiente el método analítico sintético tiene su fundamentación de aplicación en cómo vamos a tratar la información teórica necesaria para la estructuración del tema de proyecto.

3.2.3 Método Deductivo

Este método me permitió analizar la información de una forma general, el mismo que desarrolló el trabajo investigativo realizado, se inicia analizando los parámetros que conforman el desarrollo del proyecto, Desarrollar e implementar un sistema para la empresa "SERCOMEX", a fin de optimizar el proceso de ventas en la empresa para el óptimo desarrollo de la misma.

Parte de una premisa general para obtener las conclusiones de un caso particular.

Puse énfasis en la teoría, modelos teóricos, la explicación y abstracción, antes de recoger datos empíricos, hacer observaciones o emplear experimentos.

3.2.4 Método Inductivo

Este método me permitió realizar un análisis de la información desde su inicio hasta llegar a obtener información definida y estructurada, por lo tanto este método permite ir analizando las tareas y procedimientos que origina el sistema lo que desencadenaría una serie de información válida en la obtención de datos esperados a su vez los mismos irán incrementando su valor lo que logra consolidar un sistemas diseñado y estructurado con base de datos, por ende este método sí realiza un análisis de información, partiendo desde donde se origina la misma, determinando los flujos necesarios de información a ser aceptados por las interfaces gráficas de usuarios.

3.2.5 Método Sistémico

Este método me conduce a la estructuración lógica del proyecto "SISTEMA GEOGRÁFICO GEOREFERENCIADO DE GESTION DE CLIENTES" ya que tendrá que seguir un orden definido en la estructuración de cada capítulo, lo que permitirá a cualquier lector verificar paso a paso el análisis de la problemática, qué objetivo se persigue, cómo voy a aplicar mi investigación y finalmente en el desarrollo de una aplicación que será la vinculación de las partes constitutivas del proyecto y al mismo tiempo el objetivo general.

3.3 Técnicas de Recolección de la Información

3.3.1 Entrevista

Está técnica de recolección de información en mi proyecto denominado "SISTEMA GEOGRÁFICO GEOREFERENCIADO DE GESTION DE CLIENTES", para la Empresa "SERCOMEX", trasciende muchísimo ya que inicialmente se realizará el trabajo de investigación en las personas responsables del manejo de la información, los encargados del manejo del área de sistemas, los usuarios del sistema, los clientes, lo que definió una estrategia a fin de poder ir analizando ideas sobre todo de cómo ven la información la parte ejecutiva, la parte técnica, la parte operativa y como llega la información al cliente se puso mucho énfasis en poder delimitar cada una de las fases que se mencionó anteriormente cabe destacar que en el desarrollo de nuestro proyecto se tuvo que utilizar está técnica en muchas ocasiones que la entrevista lo llevará a efecto de manera personal en ocasiones telefónica, y otras utilizando medios tecnológicos de comunicación de datos (e-mail, chat, etc.), con lo que pudimos recopilar toda la información necesaria y lo que se ha plasmado en la propuesta denominada Desarrollo e Implementación de un Sistema Geográfico Georeferenciado de Gestión de Clientes.

Esta herramienta la aplicaremos en lo que se refiere a solicitar información a los directivos de la Empresa "SERCOMEX", para poder conocer sus estatutos, leyes, su fecha de fundación, preguntarles si ha existido con anterioridad este tipo de proyectos, si tienen el apoyo total de los empleados para ejercer sus funciones,



conocer si existen organizaciones que les apoyen en gestiones tanto para la Empresa como para el desarrollo intelectual de los empleados, y aspectos que se requiera a futuro, según el desarrollo de nuestro proyecto final.

Entrevista #1

Nombre Entrevistado(a): Srta. Eliana Cruz

Nombre Entrevistador: Xavier Ochoa

1. ¿Considera importante la creación de un Sistema Geográfico Georeferenciado? ¿Por qué?

Si, por qué con la creación de un Sistema Geográfico Georeferenciado como el que se propone crear, nos ayudará a terminar con el déficit de conocimientos y manejo de clientes, y demás personas que se involucren en dicho proceso.

2. ¿Cree usted que el personal del área de sistemas de su Empresa está capacitado para brindar una capacitación acorde a las necesidades del siglo XXI?

Si, ya que el personal con los cuales dispone la Empresa se han capacitado de manera correcta y constante por lo tanto se encuentran acorde con las necesidades del siglo XXI.

3. ¿Cuál es el número de empleados en la Empresa?

El número de empleados que posee la empresa es de 50 empleados aproximadamente los cuales están distribuidos en diversas áreas.

4. ¿Considera que los avances tecnológicos abren oportunidades en el campo estudiantil y laboral? ¿por qué?

Si, por que los avances tecnológicos nos ayudan en su mayoría, ya que dentro del ámbito estudiantil como laboral uno necesita de herramientas, programas, equipos que puedan ayudarnos a solucionar problemas que se pueden presentar en diario vivir como dentro de nuestros trabajos.

5. ¿Cuál es su visión al escuchar el nombre del proyecto “Sistema Geográfico Georeferenciado de Gestión de Clientes”?

Es el de destacar a nivel Empresarial un localización rápida de clientes en una forma gráfica.

3.3.1.1 Análisis de la entrevista

Pregunta #1

¿Considera importante la creación de un Sistema Geográfico Georeferenciado?
¿Por qué?

Respuesta:

Si, por qué con la creación de un Sistema Geográfico Georeferenciado como el que se propone crear, nos ayudará a terminar con el déficit de conocimientos y manejo de clientes y demás personas que se involucren en dicho proceso.

Análisis:

Con la pregunta planteada me pude dar cuenta de la necesidad de la implementación de este sistema en la Empresa, para un manejo rápido de la información.

Pregunta #2



¿Cree usted que el personal del área de sistemas de su Empresa está capacitado para brindar una capacitación acorde a las necesidades del siglo XXI?

Respuesta:

Si, ya que el personal con los cuales dispone la Empresa se han capacitado de manera correcta y constante por lo tanto se encuentran acorde con las necesidades del siglo XXI.

Análisis:

Me pude dar cuenta de los problemas que presentaría el personal técnico para la puesta en marcha del sistema, y capacitación a usuarios técnicos.

Pregunta #3

¿Cuál es el número de empleados en la Empresa?

Respuesta:

El número de empleados que posee la empresa es de 50 empleados aproximadamente los cuales están distribuidos en diversas áreas.

Análisis:

Con la elaboración de esta pregunta puedo llevar un concepto del tipo de capacitación que se daría a los empleados.

Pregunta #4

¿Considera que los avances tecnológicos abren oportunidades en el campo estudiantil y laboral? ¿Por qué?

Respuesta:

Si, por qué los avances tecnológicos nos ayudan en su mayoría, ya que dentro del ámbito estudiantil como laboral uno necesita de herramientas, programas,

equipos que puedan ayudarnos a solucionar problemas que se pueden presentar en diario vivir como dentro de nuestros trabajos.

Análisis:

Con la formulación de esta pregunta llegue a la conclusión de que los empleados de la empresa consideran muy necesario la actualización de tecnología en las empresas y centros de estudio ya que esto facilita el manejo de la información.

Pregunta #5

¿Cuál es su visión al escuchar el nombre del proyecto “Sistema Geográfico Georeferenciado de Gestión de Clientes”?

Respuesta:

Es el de destacar a nivel Empresarial una localización rápida de clientes en una forma gráfica.

Análisis:

Con la elaboración de esta pregunta me pude dar cuenta el impacto que tiene el nombre del sistema propuesto y si realmente era necesario.

3.2 Observación

Utilizaremos esta herramienta para poder observar las condiciones tecnológicas en la que se encuentra la empresa, ya sea en verificar el funcionamiento de los equipos, paredes pisos y techo que posee la organización.

Esta técnica es de mucha importancia en el proceso de validación de la información obtenida a través de todos los autores que intervienen en los procedimientos de ejecución de cada una de las tareas que se realizan, por lo tanto la validación de los mismos será a través de la Observación, poniendo mucho énfasis en los tiempos



que se ejecutan el número de individuos involucrados, los recursos ocupados y de igual manera los resultados obtenidos.

Esta técnica denominada Observación, mantiene su fundamento en la visualización práctica de los hechos, lo que en Sistemas es fundamental con la finalidad de poder discernir la información de cada uno de los actores que actúan en el sistema informático, determinar metas claras, sobre todo el origen de los datos, mantenimiento de los mismos y por ende el procesamiento de cada uno de ellos.

CAPÍTULO IV

DESARROLLO DE LA PROPUESTA

4.1 Diagnóstico Situacional

En la empresa existen estaciones de trabajo, para su uso básico, en un 100% son clones manufacturados, de arquitectura CISC, con Procesadores Intel / Dual Core de 32 Bits y 64 Bits.

Respecto a software se maneja paquetes de oficina tales como Microsoft office 2007, para navegar por la web se utiliza en un 80% Mozilla Firefox 3.6, el restante utiliza Internet Explorer 6.

Actualmente se cuenta con una red LAN de topología en forma de estrella. El servicio de Internet se lo comparte mediante configuraciones específicas de un router, conectado hacia un switch y este distribuye a todas las estaciones de trabajo. El proveedor de servicios es CNT.

El Firewall utilizado es propio de Windows, Antivirus Avira versión 10.0.0.561, y el acceso a ciertas páginas web se las restringe a través del router.

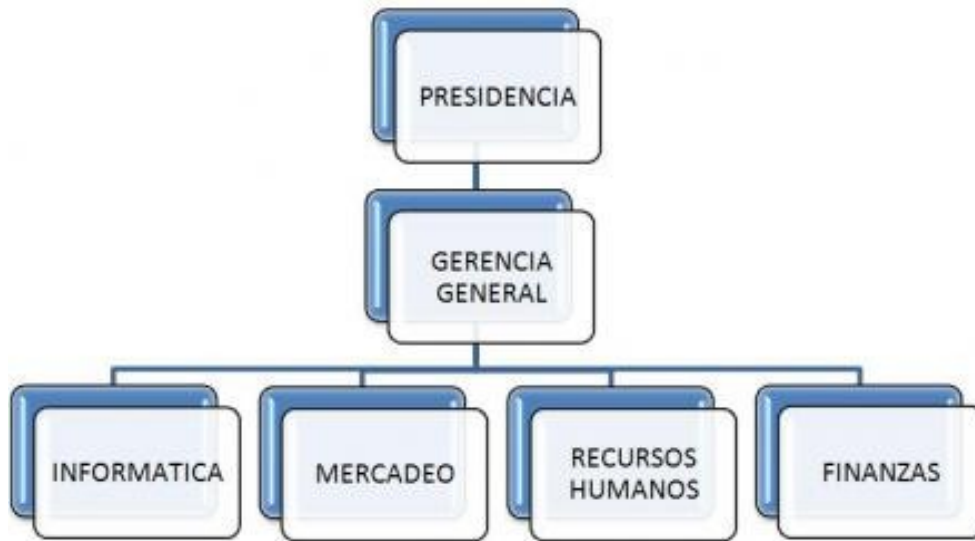
No existe ningún plan informático elaborado, que asegure o preserve la información generada, por los diferentes procesos desarrollados.

4.2 Estructura Organizacional

Es de vital importancia pues a través de esta estructura se dan a conocer los diversos puestos que existen en la institución, el nombre de cada uno de ellos, así como también el nombre de quienes ocupan cada cargo. De esta forma se determina como son las líneas de mando en la organización.

El organigrama de la Empresa “SERCOMEX”, mantiene una estructura plana, basada en funciones. Existen departamentos definidos.

Organigrama Jerárquico de la Empresa:



Estructura Organizacional
Ilustración # 5
Fuente: Empresa “SERCOMEX”

PRESIDENCIA:

El presidente de la empresa es la máxima autoridad. Se encuentra ubicado en el nivel más alto de la organización y sus funciones son:

- Dirigir y controlar el funcionamiento de la compañía.
- Representar a la compañía en todos los negocios y contratos con terceros en relación con el objeto de la sociedad
- El Presidente es el representante Legal de la Empresa, para todos los actos judiciales y extrajudiciales.

GERENTE GENERAL:

Entre sus funciones están:

- Designar todas las posiciones gerenciales.
- Realizar evaluaciones periódicas acerca del cumplimiento de las funciones de los diferentes departamentos.
- Planear y desarrollar metas a corto y largo plazo junto con objetivos claros y proyecciones de dichas metas.

- Coordinar con las oficinas administrativas para asegurar que los registros y sus análisis se están ejecutando correctamente.
- Crear y mantener buenas relaciones con los clientes para mantener el buen funcionamiento de la empresa.

DEPARTAMENTO DE INFORMATICA:

Entre sus funciones están:

- Planifica, modifica e instala nuevo software y hardware.
- Evalúa los nuevos paquetes de productos a instalarse.
- Da soporte técnico necesario para el desarrollo y evaluación del impacto de los nuevos proyectos.
- Ejecuta mantenimiento preventivo y correctivo al equipo informático de la organización.

DEPARTAMENTO DE MERCADEO:

Entre sus funciones están:

- Realiza propaganda de los servicios ofrecidos por la empresa.
- Estipula precios, ofertas y descuentos a aplicar a los servicios ofrecidos por la compañía.
- Impulsa promociones de venta.

DEPARTAMENTO DE RECURSOS HUMANOS:

Entre sus funciones están:

- Se encarga de evaluar los candidatos para un nuevo puesto de trabajo.
- Hace los procesos de contratación para el nuevo personal.
- Constata la necesidad de reclutar nuevos empleados.
- Explica y apoya a los nuevos empleados.

DEPARTAMENTO DE FINANZAS:

Entre sus funciones están:



- Declarar impuestos
- Generar balances (Inicial, final, general)
- Realizar informes de estados de resultados.
- Organizar inventarios de los activos (Edificios, mobiliario y equipo, vehículos, etc....); con los que cuenta la empresa.
- Salvaguardar los activos, y verificar si existen dichos activos.

4.3 Infraestructura Informática

4.3.1 Hardware

Características	Ubicación	Arquitectura
Intel Dual Core	Gerencia	CISC
Intel Dual Core	Secretaría	CISC
Intel Dual Core	Dirección	CISC

Hardware

Tabla # 1

Fuente: Propia del Autor

4.3.2 Software

Software	Ubicación
Office 2007	Estaciones de Trabajo, Secretaría, Dirección
Windows	Estaciones de Trabajo, Secretaría, Dirección
Mozilla Firefox	Secretaría, Dirección

Software

Tabla # 2

Fuente: Propia del Autor

4.3.3 Comunicaciones

Equipo	Ubicación
Router	Redes
Switch	Redes
Cable UTP Cat 5	Redes
Conectores RJ45	Redes
Tarjetas de Red	Redes

Comunicaciones

Tabla # 3

Fuente: Propia del Autor

4.3.4 Recurso Humano Técnico

Nombre	Descripción
Ing. Paul Curicho	Tutor de Tesis
Xavier Ochoa	Autor de Tesis
Ing. Hugo Heredia	Director Escuela de Sistemas ITSCO
Ing. Rodrigo Alegría	Director de Proyectos de Grado del ITSCO
Lic. Marcelo Meneses	Gerente de la empresa SERCOMEX

Recurso Humano Técnico

Tabla # 4

Fuente: Propia del Autor

4.4 Descripción de Alternativas

Se realiza la respectiva descripción de cada una de las alternativas propuestas para este proyecto.

Alternativa 1: PROPUESTA POR EL ESTUDIANTE DEL ITSCO

De acuerdo con las especificaciones técnicas realizadas por el alumno para el desarrollo del SISTEMA GEOGRÁFICO GEOREFERENCIADO DE GESTION DE CLIENTES podemos dar a conocer los requerimientos técnicos.

REQUERIMIENTOS TECNICOS		
DESCRIPCION	CUMPL	NO
Plataforma Windows	E X	CUMPLE
Lenguaje de programación visual 2010	X	
Base de datos SQL server 2008	X	
Interfaz grafica	X	



Módulo de Recolección de Datos	X	
Módulo de Mantenimiento	X	
Módulo de Seguridad	X	
Módulo Cartográfico	X	
Módulo de Georeferenciación	X	
Página para realizar consultas	x	
TIEMPO	4 MESES	
COSTO	\$300	
GARANTIA	1 año	
SOPORTE TÉCNICO	Inmediato	

Alternativa 1

Tabla # 5

Fuente: Propia del Autor

En la presente propuesta, presentada por el alumno estudiante del ITSCO, se hace notable el cumplimiento de todos los ítems, haciendo que la propuesta sea la mejor opción puesta en mención para el desarrollo del software.

Alternativa 2 CAAP

De acuerdo con las especificaciones técnicas solicitadas por el personal docente informático para el desarrollo del SISTEMA GEOGRÁFICO GEOREFERENCIADO DE GESTIÓN DE CLIENTES podemos dar a conocer los requerimientos técnicos.

REQUERIMIENTOS TÉCNICOS		
DESCRIPCIÓN	CUMPL	
	E	NO CUMPLE
Plataforma Windows	X	
Lenguaje de programación visual 2010	X	
Base de datos Sql server 2008	X	
Interfaz gráfica	X	
Módulo de recolección de datos	X	
Módulo de mantenimiento	X	
Módulo de seguridad	X	

Módulo de Reportes		X
Módulo Cartográfico	X	
Módulo de Georeferenciación	X	
Pagina para realizar consultas		X
TIEMPO	6 MESES	
COSTO	\$500	
GARANTIA	1 año	
SOPORTE TÉCNICO	Inmediato	

Alternativa 2

Tabla # 6

Fuente: Propia del Autor

Empresa dedicada a la creación de sistemas no cumple con los ítems Modulo de reportes y consultas, haciendo q la propuesta no sea factible.



Alternativa 3 SERCOMEX

De acuerdo con las especificaciones técnicas solicitadas por La empresa “SERCOMEX” para el desarrollo del SISTEMA GEOGRÁFICO GEOREFERENCIADO DE GESTIÓN DE CLIENTES podemos dar a conocer los requerimientos técnicos.

REQUERIMIENTOS TÉCNICOS		
DESCRIPCIÓN	CUMPL	
	E	NO CUMPLE
Plataforma Windows	X	
Lenguaje de programación visual 2010	X	
Base de datos Sql server 2008	X	
Interfaz gráfica	X	
Módulo de recolección de datos	X	
Módulo de mantenimiento	X	
Módulo de seguridad	X	
Módulo de Reportes	X	
Módulo Cartográfico	X	
Módulo de Georeferenciación	X	
Página para realizar consultas	X	
TIEMPO	6 MESES	
COSTO	\$800	
GARANTÍA	1 año	
SOPORTE TÉCNICO	Inmediato	

Alternativa 3

Tabla # 7

Fuente: Propia del Autor

La empresa **SERCOMEX** en calidad desarrolladores de sistemas cumple con la mayoría de los ítems puestos en consideración, cabe recalcar que la empresa no cumple con el ítem de la Metodología RUP.

4.5 Evaluación y Selección de Alternativas

Con la finalidad de poder determinar y seleccionar la alternativa que nos conviene a los intereses empresariales se ha definido un sistema de calificación con coeficientes matemáticos q están determinados de la siguiente manera:

Técnico 70%

Económico 20%

Garantía Técnica 5%

Soporte Técnico 5%.

TECNICO

Las especificaciones técnicas del sistema determinan la parte más importante de todo proceso de selección de alternativas escogiendo de esta forma la más idónea, por consiguiente para cada una de estas especificaciones técnicas se ha diseñado una matriz con pesos cuantitativos de acuerdo a la importancia de cada uno de los elementos que intervienen,



estos pesos sumarán una totalidad de 100 puntos lo que corresponderá el 70% de la parte técnica y mediante un cálculo matemático determinaremos lo que corresponde a cada alternativa.

ITEMS	PESOS	ALTERNATIVA 1	ALTERNATIVA 2	ALTERNATIVA 3
Plataforma: 2 capas	8	0	8	8
Metodología : Rad	8	0	0	8
Modelos: Lógico, Físico, Script de base de datos	10	10	10	10
Front End: asp.net	8	10	10	10
Back End :Sql Server 2008	8	10	10	10
Pruebas:	10	10	10	10
Unidad				
Integración				
Validación				
Sistema				
Seguridad				
Interface				
Estándares	10	0	10	10
Diseño				
Programación				
Base de datos				
Capacitación Técnica: Charlas	8	8	8	8
Capacitación Usuario: Charlas	8	8	8	8
Implantación	10	10	10	10
Manuales Técnicos	10	8	10	10
Documentación	2	2	2	2

Técnico

Tabla # 8

Fuente: Propia del Autor

ECONOMICO

A fin de determinar el puntaje respectivo que le corresponde el 20% consistente a la parte económica se define de la siguiente manera, 25 puntos cuyo costo será el más económico y progresivamente se irá asignando un costo.

El valor más caro estará determinado por la media aritmética que le corresponde a 10 puntos y de esta manera podemos determinar el valor restante que sería de 15 puntos.

De esta manera podemos calcular los valores económicos y puntajes.

Costo Económico	Costo	%
Alumno	\$0.00	25
Caap	\$300	15
Empresa "SERCOMEX"	\$ 500	10

Económico

Tabla # 9

Fuente: Propia del Autor

GARANTIA

Este ítem tiene un puntaje de 5 ya que determina el puntaje que presente el mayor tiempo de garantía, igualmente las otras alternativas se irán determinando por el tiempo.

Empresa	Garantía/Tiempo	%
Alumno	1 año	5
Docente	1 año	5
Empresa "SERCOMEX"	1 año	5

Garantía

Tabla # 10

Fuente: Propia del Autor

SOPORTE TECNICO

Está determinado por la calidad de soporte técnico que tiene la empresa, en cuanto más clientes tenga quiere decir que es una empresa con prestigio se le designara 10 puntos.

Empresa	# Visitas	%
Alumno	10	10



Docente	8	8
Empresa "SERCOMEX"	6	6

Soporte Técnico

Tabla # 11

Fuente: Propia del Autor

4.6 Factibilidad Técnica

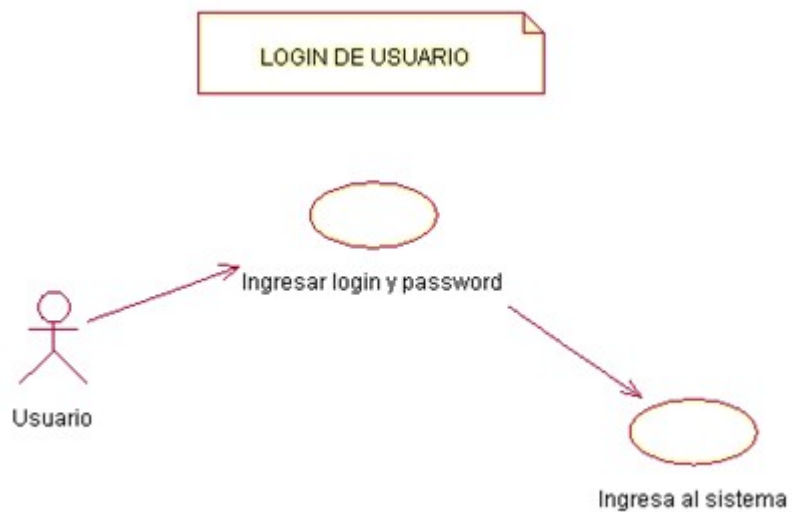
De acuerdo al análisis expuesto anteriormente en cada una de las alternativas propuestas para la solución del presente proyecto se puede concluir que su realización en la parte técnica es totalmente realizable, la misma consideración para el aspecto económico considerando en forma global que la factibilidad técnica es totalmente realizable desde el punto de vista técnico y tecnológico.

Una vez realizado el respectivo análisis de las alternativas se puede tomar la decisión importante la cual es que la alternativa N° 1 es la más conveniente.

4.7 Descripción de Procesos

4.7.1 Login de Usuarios

Seguridad en el manejo de información, ya que cada archivo tendrá que ser manipulado por personas apropiadas que van utilizar el sistema con sus respectivas copias de seguridad y así tener un buen manejo de la información.



Caso de uso: Login de Usuario

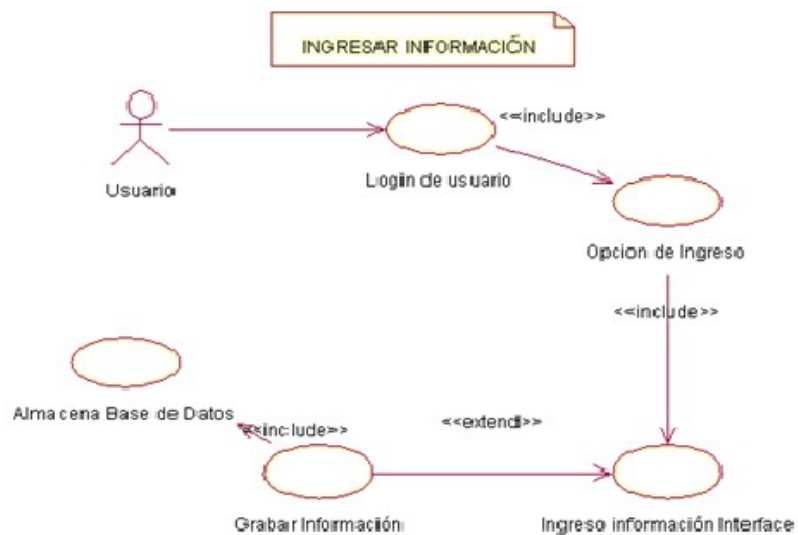
Ilustración # 6

Fuente: Rational Rose

4.7.2 Mantenimiento

Este proceso sirve para ingresar, modificar o eliminar información de los usuarios que utilizan el Sistema

4.7.2.1 Ingresar Información.



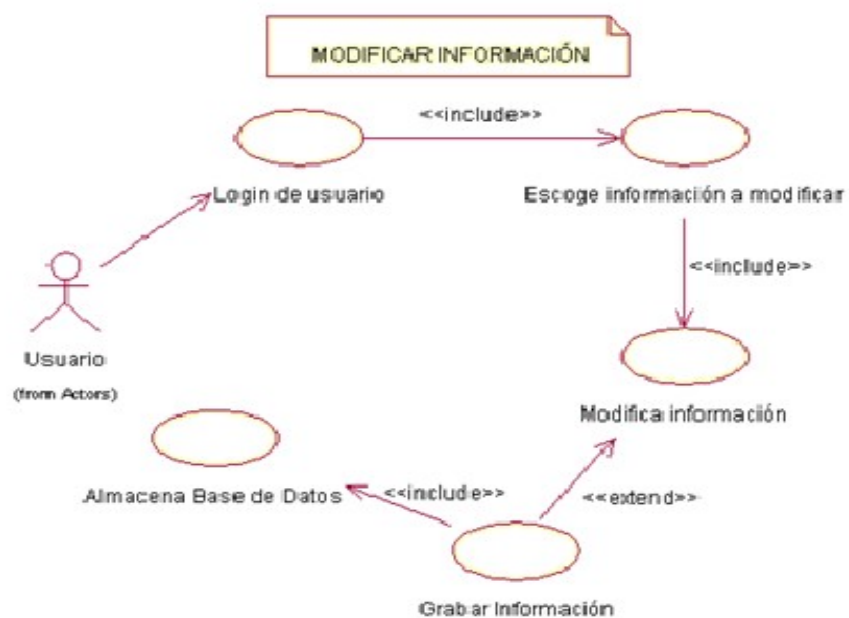
Caso de uso: Ingreso de Información

Ilustración # 7



Fuente: Rational Rose

4.7.2.2 Modificar Información.

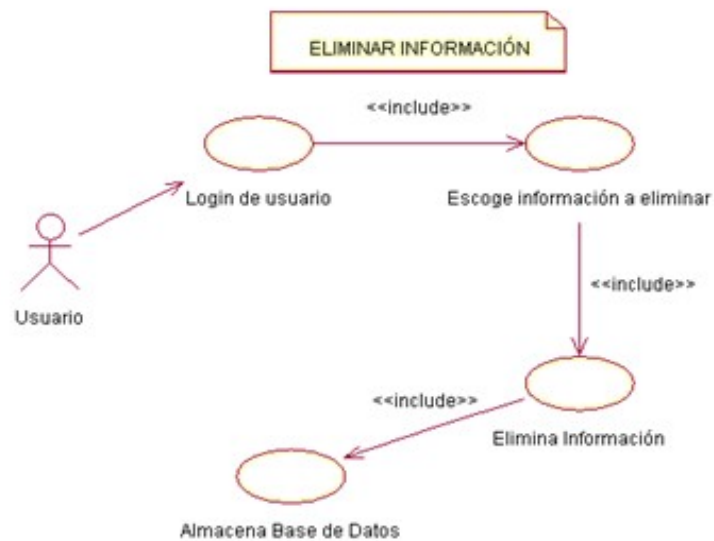


Caso de uso: Modificar Información

Ilustración # 8

Fuente: Rational Rose

4.7.2.3 Eliminar Información.



Caso de uso: Eliminar Información

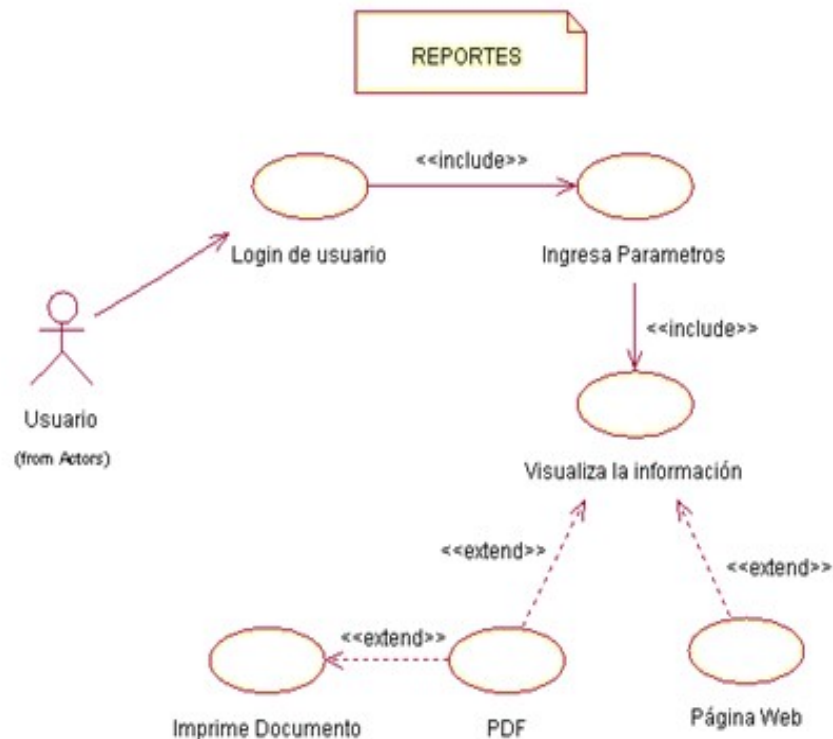
Ilustración # 9

Fuente: Rational Rose

4.7.3. Reportes

Una de las características básicas de un Sistema Geográfico Georeferenciado es su capacidad de consulta interactiva.

En este proceso, hasta la fecha, lo único que se ha podido demostrar es la capacidad de localizar de una forma gráfica y dar resultados finales de búsquedas, con la respectiva documentación necesaria para así llevar a cabo una Georeferenciación eficaz de clientes y una atención de primera calidad.



Caso de uso: Reportes
Ilustración # 10
Fuente: Rational Rose

4.8 Descripción de Metodología de Desarrollo

Para el desarrollo del sistema se utilizó la metodología RAD la misma que permitió crear el sistema en un periodo de tiempo de 90 días.

- **Planificación de los requisitos :**

En esta etapa se conoció las funciones que realiza la Empresa SERCOMEX, el proceso de Georeferenciación fue en el que se hizo énfasis ya que el sistema ha desarrollarse implica todo este proceso.

- **Etapa de diseño:**
 1. Creación de la base de datos
 2. Creación de diagramas que definen las alteraciones entre los procesos y los datos
- 3. Diseño del sistema.
- **Construcción:**
 1. Se tomo en cuenta el objetivo principal de la propuesta para el desarrollo del sistema
 2. Estándares para el desarrollo del sistema.
 3. Desarrollo y programación del software.
 4. Elaboración de manual técnico y manual de usuario.
- **Implantación:**
 1. Se realizará las pruebas correspondientes que garanticen el buen funcionamiento del sistema.
 2. Se realiza la instalación del sistema.
- **Otras actividades a tomar en cuenta son:**
 1. Capacitaciones a los empleados de la empresa.

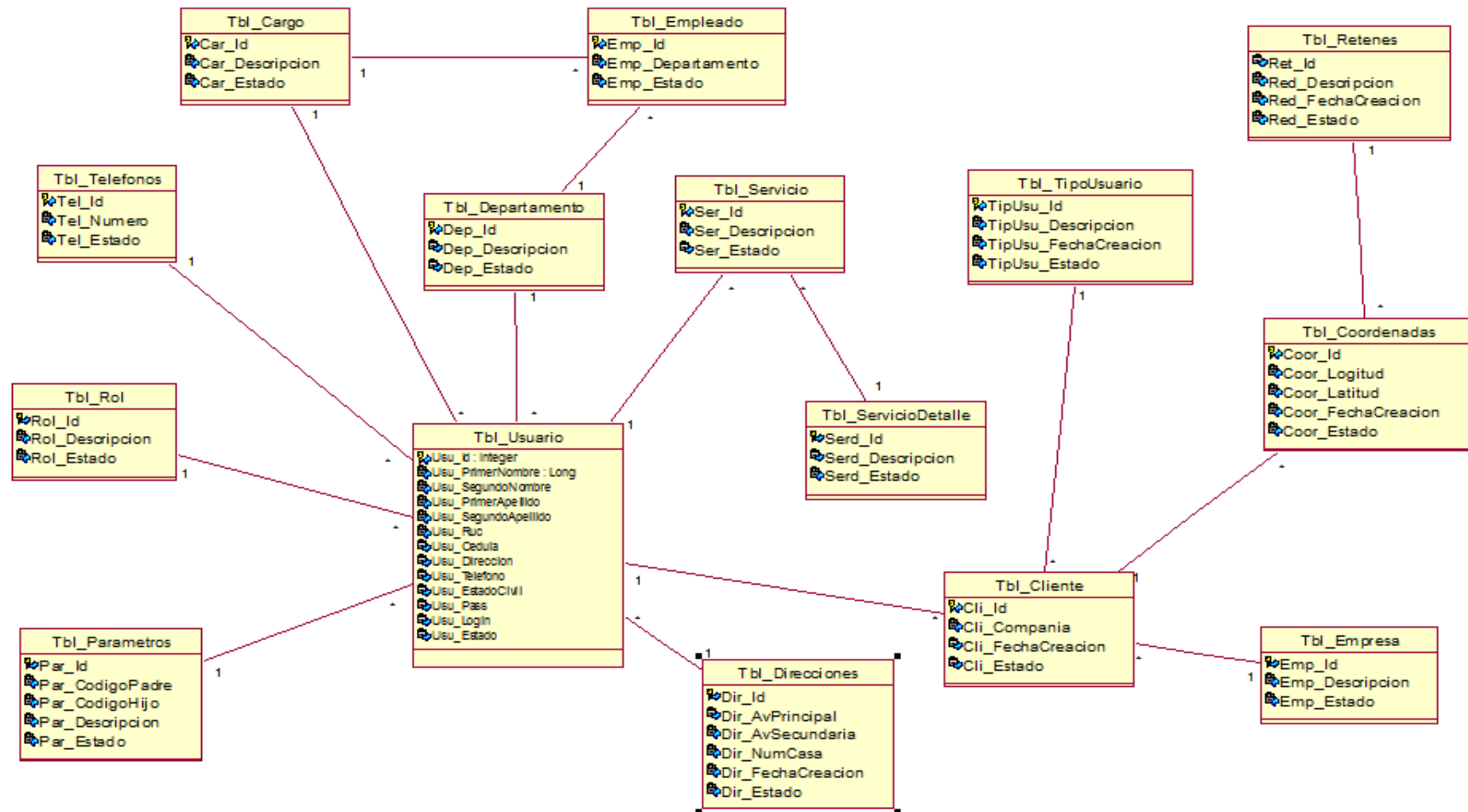
4.9 Modelo Lógico

El modelo lógico describe los datos que realmente son almacenados en la base de datos y las relaciones que existen entre los mismos, describe la base de datos completa en términos de su estructura de diseño. En el SISTEMA GEOGRAFICO GEOREFERENCIADO, hemos diseñado tablas mediante la investigación y a su vez observando las necesidades de la empresa.

La base de datos consta de las siguientes tablas: Usuario, Rol, Teléfonos, Departamento, Cargo, Empleado, Direcciones, Servicio, Detalle Servicio,



Parámetros, Clientes, Retenes, Tipo Usuario, Coordenadas, Empresa. En donde la tabla principal, es Usuarios la misma que está relacionada con la tabla Direcciones la cual nos permitirá realizar consultas de Direcciones con su mapa, otra de las tablas con mucha importancia es la de Clientes, ya que esta tabla consta con todos los Datos de cada cliente.



Modelo Logico
Ilustración # 11
Fuente: Rational Rose

4.10 Modelo Físico

El modelo físico describe cómo se almacenan los datos en el ordenador: el formato de los registros, la estructura de los ficheros (desordenados, ordenados, etc.) y los métodos de acceso utilizados (índices, etc.).

A la descripción de una base de datos se le denomina esquema de la base de datos. Este esquema se especifica durante el diseño, y no es de esperar que se modifique a menudo.

Sin embargo, los datos que se almacenan en la base de datos pueden cambiar con mucha frecuencia: se insertan datos, se actualizan, etc. Los datos que la base de datos contiene en un determinado momento se denominan estado de la base de datos u ocurrencia de la base de datos.

La distinción entre el esquema y el estado de la base de datos es muy importante. Cuando definimos una nueva base de datos, sólo especificamos su esquema al Sistema de Gestión de Base de Datos. En ese momento, el estado de la Base de Datos es el "estado vacío", sin datos. Cuando se cargan datos por primera vez, la base de datos pasa al "estado inicial", de ahí en adelante, siempre que se realice una operación de actualización de la base de datos, se tendrá un nuevo estado.

El Sistema de Gestión de Base de Datos se encarga, en parte, de garantizar que todos los estados de la Base de Datos sean estados válidos que satisfagan la estructura y las restricciones especificadas en el esquema.

Por lo tanto, es muy importante que el esquema que se especifique al Sistema de Gestión de Base de Datos sea correcto y se debe tener muchísimo cuidado al diseñarlo el mismo.



4.11 Diccionario de Datos

Un diccionario de datos contiene las características lógicas de los datos que se van a utilizar en el sistema que se está desarrollando, incluyendo nombre, descripción, alias, contenido y organización. Estos diccionarios se desarrollan durante el análisis de flujo de datos y ayuda a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño del proyecto.

Tbl_Cargo

Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	Car_Id		Integer	YES	NO		
	Car_Descripcion		Varchar (20)	NO	NO		
	Car_Estado		Char (1)	NO	NO		

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
Relationship4 (Relationship4)	Non-identifying	Tbl_Cargo	Tbl_Empleado	1:N
Relationship14 (Relationship14)	Non-identifying	Tbl_Cargo	Tbl_Usuario	1:N

Tbl_Clientes

Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
-----	------------	--------	-----------	---------	--------	-------	---------



PK	Cli_Id	Integer	YES	NO
	Cli_Compania	Varchar (20)	NO	NO
	Cli_FechaCreacion	Datetime	NO	NO
	Cli_Estado	Char (1)	NO	NO
FK	Usu_Id	Integer	YES	NO
FK	TipUsu_Id	Integer	YES	NO
FK	Emp_Id	Integer	YES	NO

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
Relationship8 (Relationship8)	Non-identifying	Tbl_Usuario	Tbl_Clientes	1:N
Relationship9 (Relationship9)	Non-identifying	Tbl_TipoUsuario	Tbl_Clientes	1:N
Relationship10 (Relationship10)	Non-identifying	Tbl_Empresa	Tbl_Clientes	1:N
Relationship18	Non-identifying	Tbl_Clientes	Tbl_Coordenadas	1:N

Tbl_Coordenadas

Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	Coor_Id		Integer	YES	NO		
	Coor_Longitud		Decimal (10,4)	NO	NO		
	Coor_Latitud		Decimal (10,4)	NO	NO		



	Coor_FechaCreacion	Datetime	NO	NO
	Coor_Estado	Char (1)	NO	NO
gFK	Ret_Id	Integer	YES	NO
FK	Cli_Id	Integer	YES	NO

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
Relationship17 (Relationship17)	Non-identifying	Tbl_Retenes	Tbl_Coordenadas	1:N
Relationship18 (Relationship18)	Non-identifying	Tbl_Clientes	Tbl_Coordenadas	1:N

Tbl_Departamento

Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	Dep_Id		Integer	YES	NO		
	Dep_Descripcion		Char (1)	NO	NO		
	Dep_Estado		Char (1)	NO	NO		

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
Relationship5	Non-identifying	Tbl_Departamento	Tbl_Empleado	1:N



(Relationship5)

Relationship13 Non-identifying Tbl_Departamento Tbl_Usuario 1:N

(Relationship13)

Tbl_Direcciones

Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	Dir_Id		Integer	YES	NO		
	Dir_AvPrincipal		Varchar (20)	NO	NO		
	Dir_AvSecundaria		Varchar (20)	NO	NO		
	Dir_NumCasa		Varchar (10)	NO	NO		
	Dir_FechaCreacion		Datetime	NO	NO		
	Dir_Estado		Char (1)	NO	NO		

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.

Tbl_Empleado

Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	Emp_Id		Integer	YES	NO		
	Emp_Departamento		Varchar (20)	NO	NO		



	Emp_Cargo	Varchar (20)	NO	NO
	Emp_Estado	Char (1)	NO	NO
FK	Car_Id	Integer	YES	NO
FK	Dep_Id	Integer	YES	NO
FK	TipUsu_Id	Integer	YES	NO
FK	Usu_Id	Integer	YES	NO

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
Relationship4 (Relationship4)	Non-identifying	Tbl_Cargo	Tbl_Empleado	1:N
Relationship5 (Relationship5)	Non-identifying	Tbl_Departamento	Tbl_Empleado	1:N
Relationship6 (Relationship6)	Non-identifying	Tbl_TipoUsuario	Tbl_Empleado	1:N
Relationship7 (Relationship7)	Non-identifying	Tbl_Usuario	Tbl_Empleado	1:N

Tbl_Empresa

Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	Emp_Id		Integer	YES	NO		
	Emp_Nombre		Varchar (20)	NO	NO		



Emp_Estado	Char (1)	NO	NO
------------	----------	----	----

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
Relationship10 (Relationship10)	Non-identifying	Tbl_Empresa	Tbl_Clientes	1:N

Tbl_Retenes

Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	Ret_Id		Integer	YES	NO		
	Ret_Descripcion		Varchar (20)	NO	NO		
	Ret_FechaCreacion		Datetime	NO	NO		
	Ret_Estado		Char (1)	NO	NO		

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
Relationship17 (Relationship17)	Non-identifying	Tbl_Retenes	Tbl_Coordenadas	1:N

Tbl_Rol



Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	Rol_Id		Integer	YES	NO		
	Rol_Descripcion		Varchar (20)	NO	NO		
	Rol_Estado		Char (1)	NO	NO		

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
Relationship16 (Relationship16)	Non-identifying	Tbl_Rol	Tbl_Usuario	1:N

Tbl_Servicio

Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	Ser_Id		Integer	YES	NO		
	Ser_Detalle		Varchar (20)	NO	NO		
	Ser_Estado		Char (1)	NO	NO		
FK	TipUsu_Id		Integer	YES	NO		
FK	Usu_Id		Integer	YES	NO		
FK	Serd_Id		Integer	YES	NO		

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
--------------------------------	------------------	-------------	------------	-------



Relationship11 (Relationship11)	Non-identifying	Tbl_TipoUsuario	Tbl_Servicio	1:N
Relationship12 (Relationship12)	Non-identifying	Tbl_Usuario	Tbl_Servicio	1:N
Relationship19 (Relationship19)	Non-identifying	Tbl_ServicioDetalle	Tbl_Servicio	1:N

Tbl_ServicioDetalle

Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	Serd_Id		Integer	YES	NO		
	Serd_Estado		Char (1)	NO	NO		

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
Relationship19 (Relationship19)	Non-identifying	Tbl_ServicioDetalle	Tbl_Servicio	1:N

Tbl_Telefonos

Columns



Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	Tel_Id		Integer	YES	NO		
	Tel_Numero		Varchar (10)	NO	NO		
	Tel_Estado		Char (1)	NO	NO		

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
Relationship20 (Relationship20)	Non-identifying	Tbl_Telefonos	Tbl_Usuario	1:N

Tbl_TipoUsuario

Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	TipUsu_Id		Integer	YES	NO		
	TipoUsu_Descripcion		Varchar (20)	NO	NO		
	TipoUsu_FechaCreacion		Datetime	NO	NO		
	TipoUsu_Estado		Char (1)	NO	NO		

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
--------------------------------	------------------	-------------	------------	-------



Relationship6 (Relationship6)	Non-identifying	Tbl_TipoUsuario	Tbl_Empleado	1:N
Relationship9 (Relationship9)	Non-identifying	Tbl_TipoUsuario	Tbl_Clientes	1:N
Relationship11 (Relationship11)	Non-identifying	Tbl_TipoUsuario	Tbl_Servicio	1:N

Tbl_Usuario

Columns

Key	Columnname	Domain	Data type	Notnull	Unique	Check	Default
PK	Usu_Id		Integer	YES	NO		
	Usu_PrimerNombre		Varchar (20)	NO	NO		
	Usu_SegundoNombre		Varchar (20)	NO	NO		
	Usu_PrimerApellido		Varchar (20)	NO	NO		
	Usu_SegundoApellido		Varchar (20)	NO	NO		
	Usu_NombreUsuario		Varchar (20)	NO	NO		
	Usu_Ruc		Numeric (12,0)	NO	NO		
	Usu_RazonSocial		Varchar (20)	NO	NO		
	Usu_Estado		Char (1)	NO	NO		
FK	Dep_Id		Integer	YES	NO		
FK	Car_Id		Integer	YES	NO		
FK	Rol_Id		Integer	YES	NO		
FK	Tel_Id		Integer	YES	NO		



FK	Dir_Id	Integer	YES	NO
----	--------	---------	-----	----

Relationships

Constraint (relationship) name	Relationshiptype	Parenttable	Childtable	Card.
Relationship7 (Relationship7)	Non-identifying	Tbl_Usuario	Tbl_Empleado	1:N
Relationship8 (Relationship8)	Non-identifying	Tbl_Usuario	Tbl_Clientes	1:N
Relationship12 (Relationship12)	Non-identifying	Tbl_Usuario	Tbl_Servicio	1:N
Relationship13 (Relationship13)	Non-identifying	Tbl_Departamento	Tbl_Usuario	1:N
Relationship14 (Relationship14)	Non-identifying	Tbl_Cargo	Tbl_Usuario	1:N
Relationship16 (Relationship16)	Non-identifying	Tbl_Rol	Tbl_Usuario	1:N
Relationship20 (Relationship20)	Non-identifying	Tbl_Telefonos	Tbl_Usuario	1:N
Relationship21 (Relationship21)	Non-identifying	Tbl_Direcciones	Tbl_Usuario	1:N

4.12 Estándares de Programación



5 4.12.1 Nomenclatura

El nombre de los objetos de base de datos no deberá exceder los 35 caracteres y no deberá terminar en “_”. En el caso que el nombre esté compuesto por más de una palabra, las mismas tendrán que separarse por el signo “_” underscore.

Los nombres de objetos de base de datos deberán estar escritos en español, ser auto-descriptivos (nombres completos). El Área de involucramiento técnico asignará el nombre del proyecto, nombre de esquema y el prefijo de la aplicación a utilizarse en la nomenclatura de los objetos de base de datos, a fin de dar identidad al proceso de seguimiento y rastreo vehicular.

6 4.12.1.1 Documentación

Todos los objetos de base de datos y scripts deberán estar debidamente documentados, en las siguientes propiedades:

- **Comentario:** descripción clara del uso o funcionalidad del objeto. Cabe notar que esta documentación se verá reflejada en la base de datos.
- **Descripción:** este campo es opcional si el objeto tiene el campo comentarios y contendrá aclaraciones o ejemplos. Caso contrario es obligatorio y debe contener la descripción detallada del uso o funcionalidad del objeto.
- **Notas:** este campo se utilizará exclusivamente en el caso de requerirse la eliminación del objeto y contendrá los siguientes puntos:
 - Eliminación solicitada por:
 - Fecha de Eliminación:
 - Motivo de Eliminación:
 - Solución de Reemplazo: en el caso que aplique se debe detallar que solución reemplaza a la funcionalidad brindada por el objeto a ser eliminado.



Es importante que la documentación permita entender claramente el propósito del objeto y no que sea una simple repetición del nombre.

7 Tipos de Datos

Tipo de Dato	Cuando se debe utilizar
VARCHAR2	Para campos de texto de tamaño variable de hasta 5000 caracteres. Siempre se utilizará VARCHAR2 en lugar de VARCHAR.
CHAR	Para campos de texto de tamaño fijo, por ejemplo para el uso de estados (SI/NO), (ACT/INA).
NUMBER	Para campos numéricos. Siempre se especificará la precisión, por ejemplo para un número de 5 cifras enteras y 2 decimales será: NUMBER (7,2).
DATE	Para campos de fecha y de fecha y hora.
CLOB	Para campos de texto de más de 5000 caracteres.
BLOB	Para almacenamiento de archivos binarios, por ejemplo imágenes, archivos pdf, Word, etc.
XMLTYPE	Para campos en formato XML sobre los cuales se ejecutarán consultas con XPATH desde la Base de Datos. Si no se requerirá utilizar el motor de XML de la Base de Datos, es preferible utilizar un campo CLOB

Tipo de Datos

Tabla # 12

Fuente: Propia del Autor

Nombre:	[Prefijo de la aplicación] “_” [Nombre auto-descriptivo] Para roles de Discoverer: [Prefijo de la aplicación] “_DIS_” [Nombre auto-descriptivo]
---------	---



Documentación	<p>Comments: incluirá el perfil del usuario que utilizará el rol y de manera general la funcionalidad a la que tendrá acceso, por ejemplo:</p> <p>Para el rol RIG_AIR_GESTION:</p> <p>Rol utilizado por usuarios que realizan gestión sobre la presentación del anexo de otras retenciones.</p> <p>Permite la consulta de presentaciones de anexos realizadas por los contribuyentes.</p>
Consideración:	<p>Toda aplicación deberá tener el rol:</p> <p>[Prefijo de la aplicación]_MANTENIMIENTO con permisos de select a todas las tablas del esquema de la aplicación.</p>

8 4.12.1.2 Tablas

Nombre:	<p>[Prefijo de la aplicación] “_” [Nombre auto-descriptivo en “singular”]</p> <p>De manera general el nombre auto-descriptivo de una tabla de rompimiento se conformará con la unión del nombre de las dos tablas padres.</p>
Alias:	<p>Se utilizará el alias generado por Designer. Si ya se encuentra en uso el prefijo asignado, se añadirá al final un número secuencial de 1 a 9.</p>
Consideraciones:	<p>Toda tabla tendrá una “clave primaria numérica de un solo campo” salvo mejor criterio de Diseño.</p>



Estándares de Tablas

Tabla # 13

Fuente: Propia del Autor

8.1.1.1 Clave Primaria (Primary Key)

Nombre Campo:	"CODIGO_" [Nombre de la tabla en singular, sin prefijo de la aplicación, hasta donde alcance]
Nombre PK:	Utilizar el nombre asignado por Designer ([alias de la tabla] "_PK")
Consideraciones:	<p>La clave primaria debe ser numérica y de un solo campo, salvo mejor criterio de Diseño.</p> <p>Como caso de excepción se permitirá la creación de claves primarias compuestas, en el caso en que la mejor representación a nivel de entidades en un diagrama de clases sea una relación de muchos a muchos y:</p> <ul style="list-style-type: none">➤ No se requiere registrar información adicional en la tabla de rompimiento➤ No se requiere relacionar la tabla de rompimiento con ninguna otra tabla.
Documentación:	Se requiere únicamente la documentación del campo en la tabla (no se requiere documentar el primarykey). En el caso de utilizar una secuencia, se colocará la referencia a la misma utilizando la sección SEQUENCE de Designer.

Estándares de Campos Primarios

Tabla # 14

Fuente: Propia del Autor



8.1.1.2 Campos

Nombre:	<p>El nombre de un campo deberá ser auto-descriptivo y estar en singular.</p> <p>Únicamente la clave primaria y los campos que provienen de un foreignkey pueden comenzar con la palabra CODIGO.</p> <p>El nombre no debe incluir valores permitidos, por ejemplo: ESTADO_ACTIVO_INACTIVO, lo correcto sería: ESTADO.</p>
Tipo de Dato:	Revisar las consideraciones del punto “ ”
Consideraciones:	<p>Todo campo con un conjunto de valores limitados que:</p> <p>a) Varían en el tiempo, harán referencia a la tabla o replica de la tabla correspondiente. Por ejemplo ID_VEHICULO debe tener un foreignkey hacia EVENTO, etc.</p> <p>Diseño analizará los casos de excepción en los que no es posible crear foreignkeys, por ejemplo:</p> <p>Tablas no replicadas.</p> <p>Tablas replicadas con tipo COMPLETE</p> <p>En tablas de log.</p> <p>b) Son fijos, tendrán obligatoriamente allowablevalues registrados en Designer de la siguiente manera:</p> <p>Value: tres letras representativas de cada palabra, por ejemplo:</p> <p>Para ACTIVO, INACTIVO, serían ACT e INA.</p> <p>Para ACEPTADO_MANUALMENTE y ACEPTADO_AUTOMATICAMENTE serían ACE_MAN y ACE_AUT.</p>



	<p>Para valores con menos de tres letras se colocará el nombre completo, por ejemplo: SI, NO, 0+.</p> <p>Meaning: nombre completo del valor permitido, por ejemplo ACTIVO, INACTIVO, etc.</p> <p>Caso de Excepción para MDA: en el caso de aplicaciones que cuenten con un modelo UML los allowablevalues deberán ser diseñados en UML a través de Enumeraciones y se cargarán en Designer como checkconstraints en lugar de allowablevalues. La documentación del meaning deberá ser incluida en el comment del checkconstraint, ejemplo: "ACT = Activo, INA = Inactivo"</p>
Documentación:	<p>Comentario: se debe incluir la utilidad del campo y los casos de uso, por ejemplo:</p> <p>Código de la Agencia principal del SRI, en la cual se reciben los trámites que llegan de otras oficinas</p> <p>Descripción: este campo es opcional y debe incluir aclaraciones y ejemplos respecto al contenido del campo.</p> <p>En el caso de utilizar una secuencia, se colocará la referencia a la misma utilizando la sección SEQUENCE de Designer.</p>

Estándares de Campos

Tabla # 15

Fuente: Propia del Autor

Tablas

Reglas generales



- Los nombres deben especificarse en plural, y de acuerdo a las reglas generales.

Ejemplos: departamentos, facturas, monedas.

- En el caso de tablas que se relacionan específicamente con otra tabla (ej. tablas tipo,

Nomencladores, entidades débiles), esta relación debe quedar expresada en el nombre.

Ejemplos: domicilios Personas, categorías Municipios.

- Las tablas de relación (objetos asociativos, representan relaciones de N a M) deben nombrarse utilizando los nombres de las tablas intervinientes, siguiendo un orden lógico de frase.

Ejemplos: localidades Municipios, facturas Notas

Campos clave (Identificadores de tabla)

- Toda tabla debe poseer uno o más campos clave.
- Toda relación entre tablas debe implementarse mediante constraints (claves foráneas) con integridad referencial, de acuerdo al motor de base de datos utilizado.



- La integridad referencial deberá actualizar en cascada en todos los casos, y restringir el borrado salvo para las entidades débiles.

Ejemplos: no se podrá eliminar un registro de la tabla localidades que tenga ocurrencias en otras tablas; para este caso deberá implementarse el borrado lógico.

- Los campos clave deben ubicarse al inicio de la definición de la tabla (deben ser los primeros).
- El nombre del campo clave debe estar compuesto por “id” + nombre de la tabla en singular.

Dependiendo de la naturaleza de la entidad, el nombre de la tabla a usar es el de la misma tabla, o el de la relacionada.

Ejemplos: tabla localidades => idLocalidad.

- Las claves compuestas sólo deben utilizarse en casos específicos, por ejemplo, tablas de relación o entidades débiles, deberá generarse un campo clave y generar un índice único en los campos que la identificaban.

Otros campos

- Todo campo que represente un nombre o descripción, se colocará inmediatamente después de los campos clave, y se nombrará como a la tabla a la que pertenece, en singular.



Ejemplos: tabla localidades => idLocalidad, localidad.

- Algunos campos que representan datos, de acuerdo a su representación conceptual en el ámbito del negocio, deberán prefijarse de la siguiente manera:

Números: num (ejemplo: Número de factura => numFactura)

Fechas: fecha (ejemplo: Fecha de inscripción => fechaInscripcion)

Códigos: codigo (ejemplo: Código de producto: codigoProducto)

- Los campos booleanos deberán nombrarse de acuerdo al estado correspondiente al valor 1/Verdadero/True de los mismos.

Ejemplos: autorizado, oculto, vigente.

- Los campos de relación (foreign keys, claves foráneas) deben nombrarse de la misma manera que los campos clave (usando el nombre de la tabla a la que hacen referencia).

Ejemplos: tabla personas => idTipoDocumento, idEstadoCivilEstándares.

- Si el framework de desarrollo utilizado genera o requiere nombres de campos con nomenclatura específica

Pautas para el nombramiento de una clase:

- El nombre de la clase debe de iniciar con letra Mayúscula.



- Si el nombre de la clase contiene más de una palabra se juntarán sin dejar espacio en blanco, el inicio de cada palabra se escribirá con letra Mayúscula.

Pautas para el nombramiento de atributos y operaciones o métodos:

- El nombre de los atributos y las operaciones deben iniciar con letra minúscula.
- Si el nombre consta de más de una palabra colocarlas juntas sin dejar espacio, al inicio de la segunda y posteriores palabras colocarlas al inicio con letra mayúscula.

Pautas para las asociaciones o relaciones:

- Se emplearán flechas que indican la relación de dos clases, la dirección de la flecha indica el sentido de la relación con sus respectivos nombres.

Las relaciones presentan multiplicidad.

4.13 Pantallas Y Reportes (Prototipo)

INGRESO AL SISTEMA

Para ingresar el sistema se da doble click en el escritorio en el icono que representa a los navegadores en este caso Explorer





Ingreso al Sistema.

Ilustración # 13

Fuente: Propiedad del Autor

Al ingresar al sistema se presenta la siguiente pantalla:

LOGIN

Esta pantalla es la de inicio del sistema, aquí nos pedirá ingresar el nombre de usuario y la contraseña, si son correctos presenta el menú del sistema, caso contrario no nos permitirá ingresar.

INICIAR SESIÓN
Especifique su nombre de usuario y contraseña.

Información de cuenta

Nombre de usuario:

Contraseña:

☐ Mantenerme conectado

Pantalla de Inicio de Sesión.

Ilustración # 14

Fuente: Propiedad del Autor



Iniciar sesión

Una vez ingresado los datos en sus respectivos campos presionamos si el
password es incorrecto se muestra el siguiente mensaje

El intento de conexión no fue correcto. Inténtelo de nuevo.

Información de cuenta

Nombre de usuario: *

Contraseña:

☐ Mantenerme conectado

Iniciar sesión

Ilustración # 15

Fuente: Propiedad del Autor

MANTENIMIENTOS



Una vez dentro, el sistema muestra la pantalla principal del sistema, esta nos presentara el siguiente formulario con la lista de usuarios, donde podemos eliminar y modificar los registros.

USUARIOS

+ NUEVO

LISTA DE USUARIOS

BUSCAR POR:

	CEDULA	NOMBRE	PRIMER APELLIDO	SEGUNDO APELLIDO
 	1234567890	Juan	Gallegos	Lara

Formulario Lista de Usuarios.

Ilustración # 16

Fuente: Propiedad del Autor



El siguiente Formulario nos muestra como ingresar datos del Usuario además de la opción de guardar los datos o crear uno nuevo

USUARIOS

+ NUEVO GUARDAR

NOMBRE:	Carlos	SEGUNDO APELLIDO	Cedeño
PRIMER APELLIDO	Aguilar	TELEFONO	2428996
CEDULA	1724596834	CARGO	Jefe Sistemas
DEPARTAMENTO	Sistemas	TIPO USUARIO	Administrador
ROL	SubJefe		

LISTA DE USUARIOS

Mantenimiento de Usuarios.

Ilustración # 17

Fuente: Propiedad del Autor

El siguiente Formulario nos muestra como Modificar datos del Usuario además de la opción de guardar los datos modificados.

USUARIOS

+ NUEVO GUARDAR

NOMBRE:	Carlos	SEGUNDO APELLIDO	Cedeño
PRIMER APELLIDO	Aguilar	TELEFONO	2428996
CEDULA	1724596834	CARGO	Jefe Sistemas
DEPARTAMENTO	Sistemas	TIPO USUARIO	Administrador
ROL	SubJefe		

LISTA DE USUARIOS

Modificar Usuarios.

Ilustración # 18

Fuente: Propiedad del Autor

BUSQUEDA



En este formulario podemos buscar al cliente de la empresa y los retenes de policía cercanos respectivamente para ello podrán hacerlo por dirección o nombre del cliente.


Tipo de búsqueda:

☒ Todos

Nombre:

Nombre de usuario:

Password:



Busqueda de Usuarios .

Ilustración # 19

Fuente: Propiedad del Autor

4.14 Pruebas y Depuración

4.14.1 Pruebas de Unidad

- Se concentra en la verificación de la unidad más pequeña del diseño del software: el componente o módulo del software.

La verificación en el diseño se realizó determinando que exista las relaciones entre cada una de las tablas, poniendo especial énfasis en las tablas que contienen la información que almacena los datos de los clientes, así como también la información de la cartografía georeferenciada.

- Las pruebas de unidad se concentran en la lógica del negocio los procesos internos que relacionan las acciones de Comando y Control que se consideren con la finalidad que la relación entre las tablas y almacenamientos de información pueda mantener una estructura adecuada y lógica.



- Se considero además que se mantenga la integridad de la información y sus datos, poniendo especial énfasis en las relaciones claves primarias y foráneas del modelo lógico del negocio.
- Este tipo de prueba se puede aplicar en paralelo a varios componentes, se realizo verificando el funcionamiento de los componentes geográficos con los componentes transaccionales, y georeferenciamiento de información, determinando la integración de cada uno de ellos.

4.14.2 Pruebas de Integración

- Esta prueba la realice individualmente verificando que todo funciona bien individualmente, por lo tanto las relaciones existentes en el modelo establecido, se verifico la consistencia del modelo indicando las claves primarias y foráneas establecidas, a fin de poder verificar la indexación de las búsquedas de las direcciones.
- La prueba de integración es una técnica sistemática para construir la arquitectura del software, mientras que al mismo tiempo, se aplican las pruebas para descubrir errores asociados con la interfaz.

La integración de los datos en la tabla de los clientes, determina la robustez del modelo colocando especial énfasis en las asociaciones de las tablas que tienen relación con la información que maneja los datos georeferenciados.

4.14.3 Pruebas de Validación

- Las pruebas de validación empiezan tras la culminación de la prueba de integración, Se ha terminado de ensamblar el software como paquete y se han descubierto y corregido los errores de interfaz. Este tipo de pruebas se la realizo en el



módulo de seguridad establecido para poder controlar el acceso de los usuarios al sistema.

- La prueba se concentra en las acciones visibles para el usuario y en la salida del sistema que éste puede reconocer. Este tipo de prueba se realizó con el módulo de seguridad poniendo énfasis en los tipos de usuarios que tiene que manejar el sistema.
- Se verificó que existan mensajes de verificación de usuarios al sistema, a través de ventanas de alertas y de precaución. La validación del software se logra mediante una serie de pruebas que demuestren que se cumple los requisitos.

Las pruebas realizadas determinan que cumple con los requisitos ya que el sistema ejecuta filtros y criterios de búsqueda razonables y efectivos a fin de poder ubicar al cliente en el mapa georeferenciado.

4.14.4 Pruebas del Sistema

- Al final del desarrollo el software se incorpora a otros elementos del sistema (hardware, personas, información) y se realiza una serie de pruebas de integración del sistema y de validación.

La carga de datos en el sistema constituye un factor determinante especialmente al realizar la migración de los datos de los clientes, por lo tanto la integración con la consola de la misma es un factor determinante a fin de poder establecer un sistema robusto y sólido en el manejo de la información, considerando que los datos de latitud y longitud servirá en la posición exacta del cliente y por consiguiente la toma de decisiones será oportuna y viable.

- Sin embargo, los pasos dados durante el diseño y la prueba del software mejorarán en gran medida la probabilidad de tener éxito en la integración del software.



La integración de los datos y las relaciones entre cada uno de las tablas se verifico la consistencia de los datos y la integridad de los mismos.

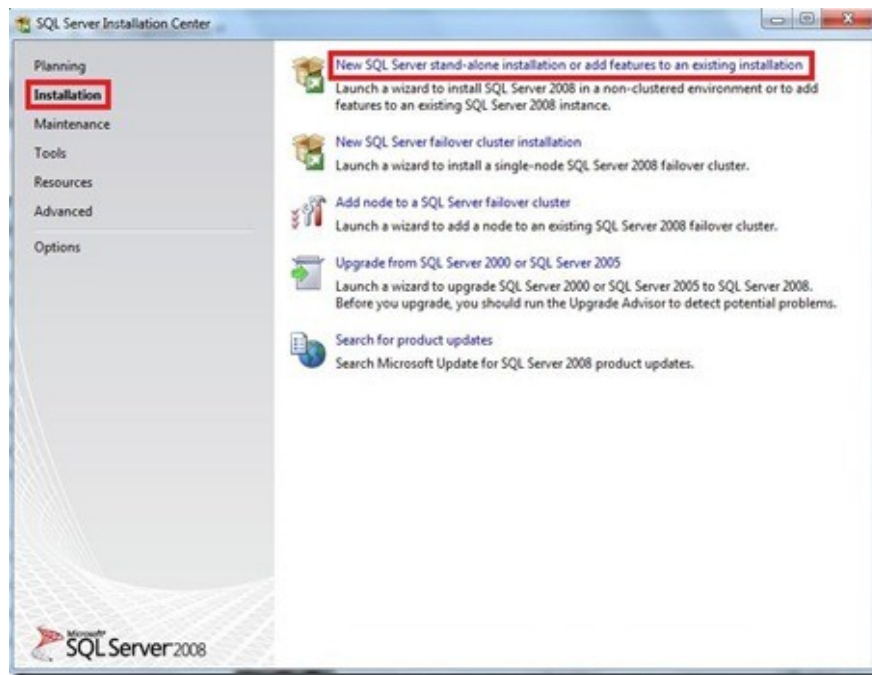
4.14.1 Prueba de Interfaz Grafica

- Se realizo una prueba con la ejecución del manejo de los iconos de acceso al sistema y la integración con la base de datos y el sistema de información geográfica, la lista de clientes en los reportes respectivos.
- El ingreso de los datos se realizara por medio de los datos del cliente como la cedula el mismo que es único para cada uno de los registros, además para cada uno de los clientes se determinará una clave de acceso individual.

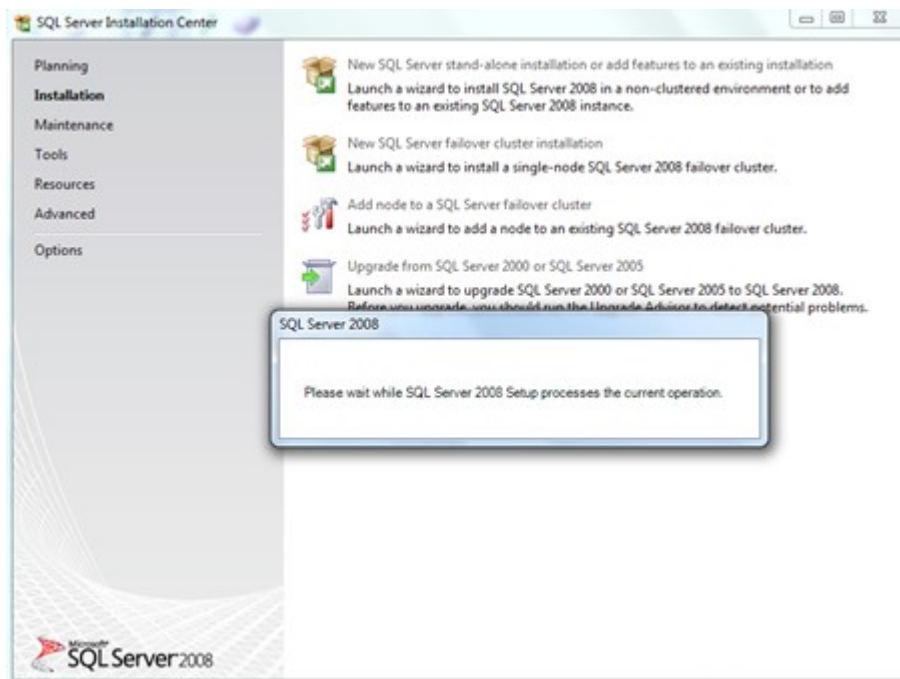
4.15 INSTALACION DEL SISTEMA

4.15.1 Instalación de SQL Server

- se nos presenta una pantalla con varias opciones, la que nos interesa es la instalación, por lo que en el menú lateral escogemos “Instalación” y luego Instalar SQL Server ó agregar características en una instalación existente.

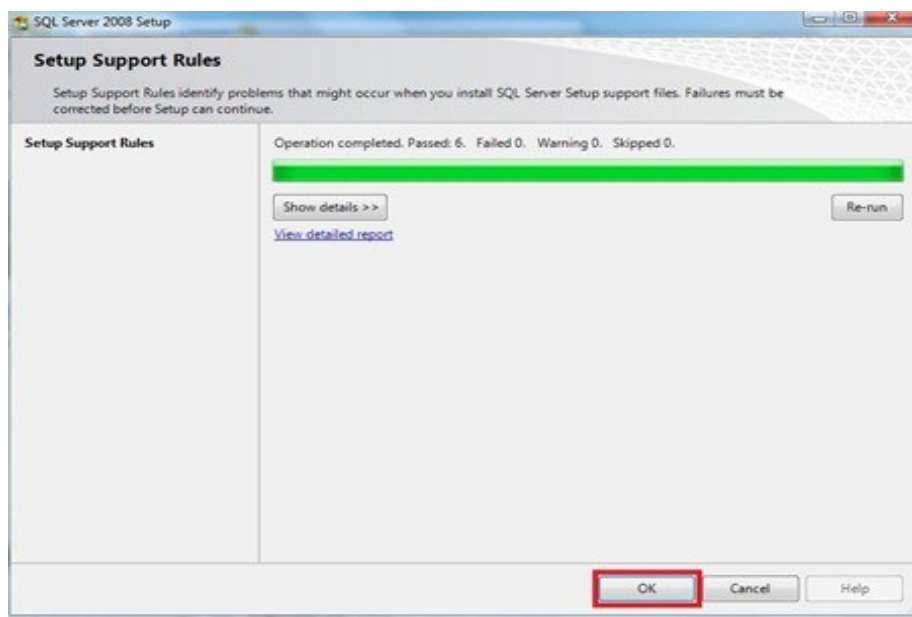


- ahora bien se nos presenta un modal en donde recibimos un mensaje de espera que nos indica que se están realizando las operaciones solicitadas

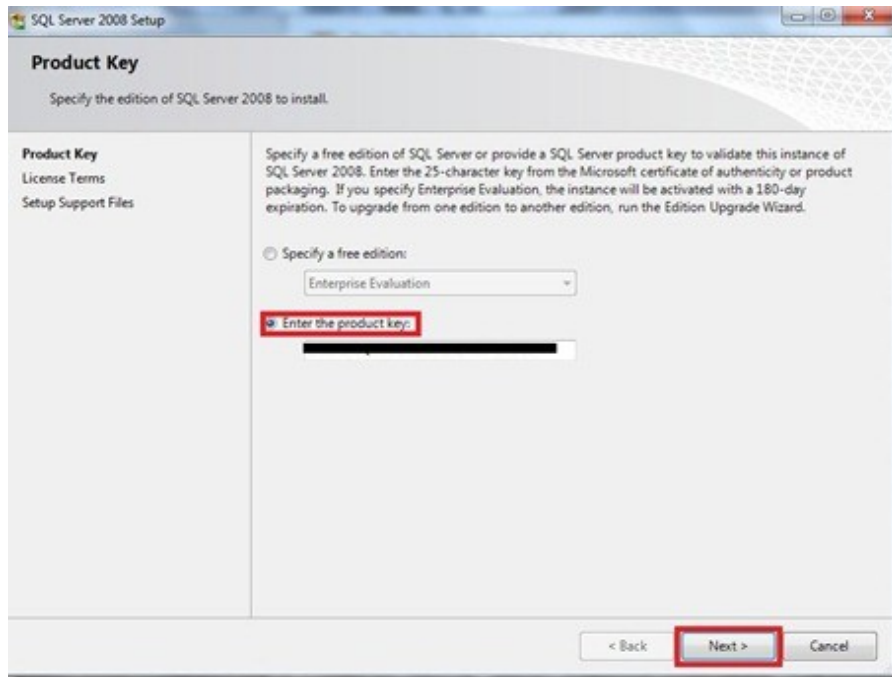




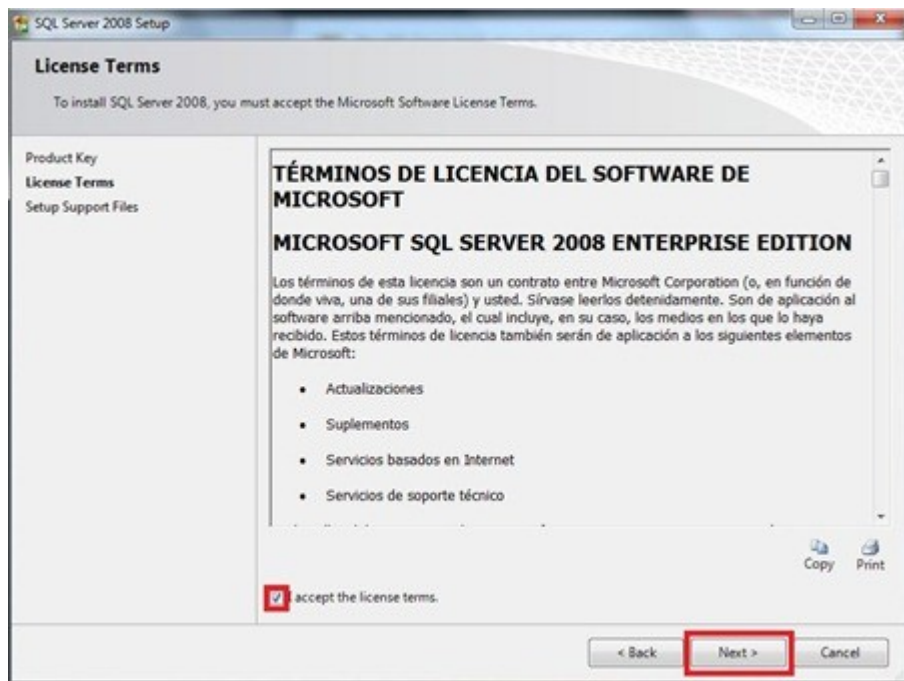
- Debemos instalar las reglas de soporte, en donde verifica si existen aplicaciones o componentes que eviten la instalación de SQL Server 2008, si este no es el caso simplemente esperamos y recibimos un mensaje que “6 operaciones se completaron exitosamente” y clic en aceptar



- En la siguiente pantalla debemos seleccionar “Introducir la llave del producto”, revisemos en el reverso del paquete de SQL Server 2008 el KEY que trae y lo introducimos finalmente clic en “aceptar”

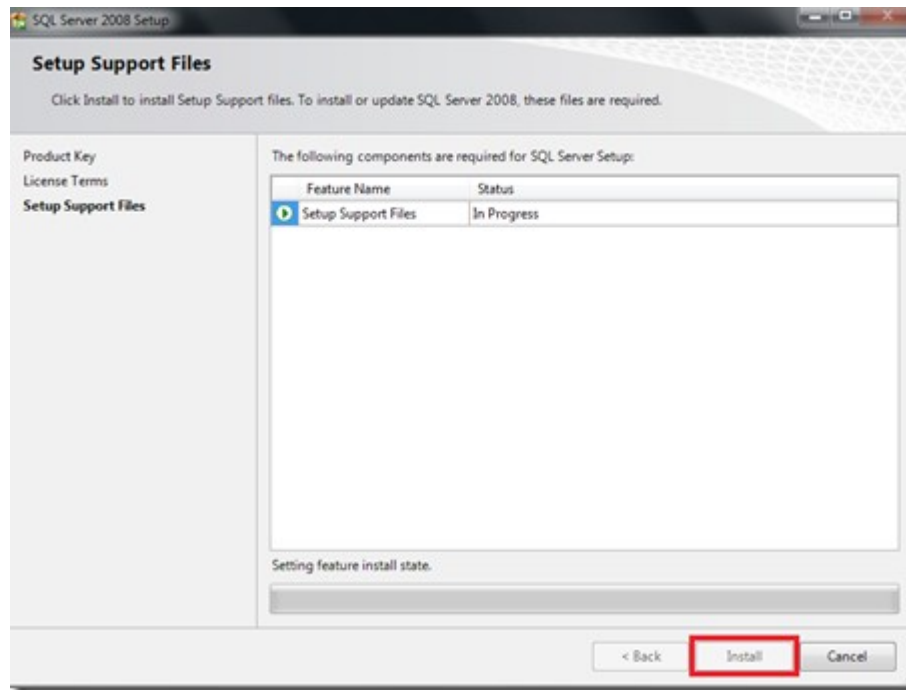


- La próxima ventana nos trae los términos de uso de licencia de software, les recomiendo que lo lean y si están de acuerdo clic en “Siguiente”

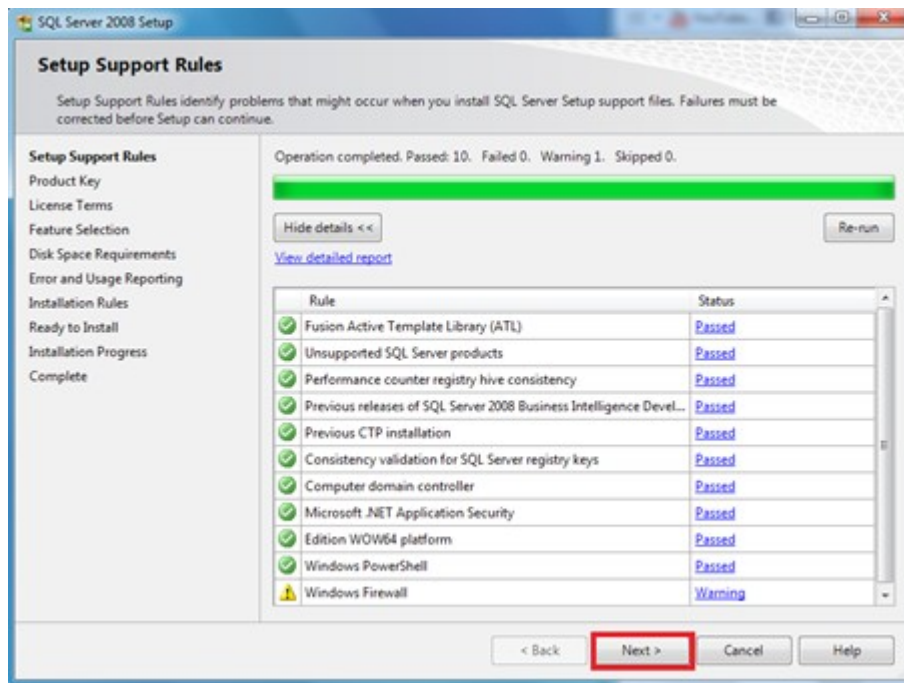




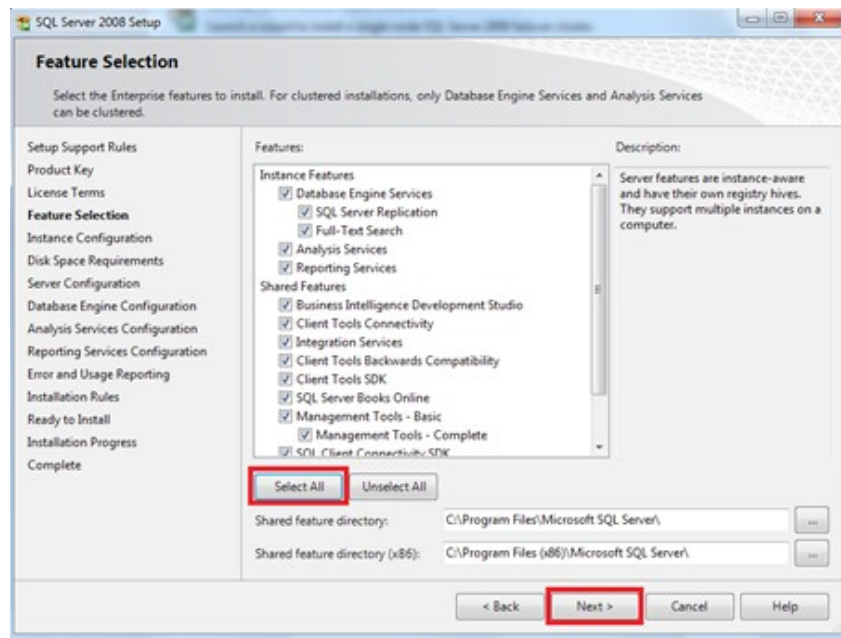
- instalamos ahora los archivos de soporte que serán nuestra guía durante la configuración de características de SQL Server 2008, simplemente seleccionamos “Instalar”



- Cuando haya terminado la instalación de archivos de soporte, tendremos un resumen de las características que fueran agregadas y seleccionamos “Siguiente”

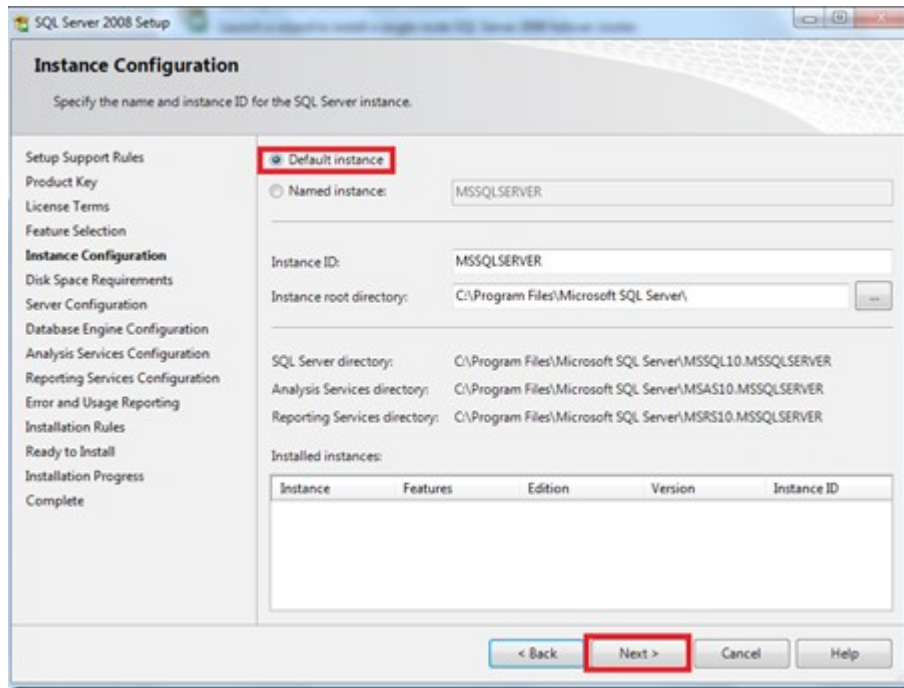


- esta es la parte más importante de la instalación en donde el usuario deberá seleccionar que características desea instalar para su equipo, desde el motor de base de datos hasta reportes, en mi caso quiero todos los componentes en mi PC, por lo que elijo “Seleccionar todos” y a ahora “Siguiente”

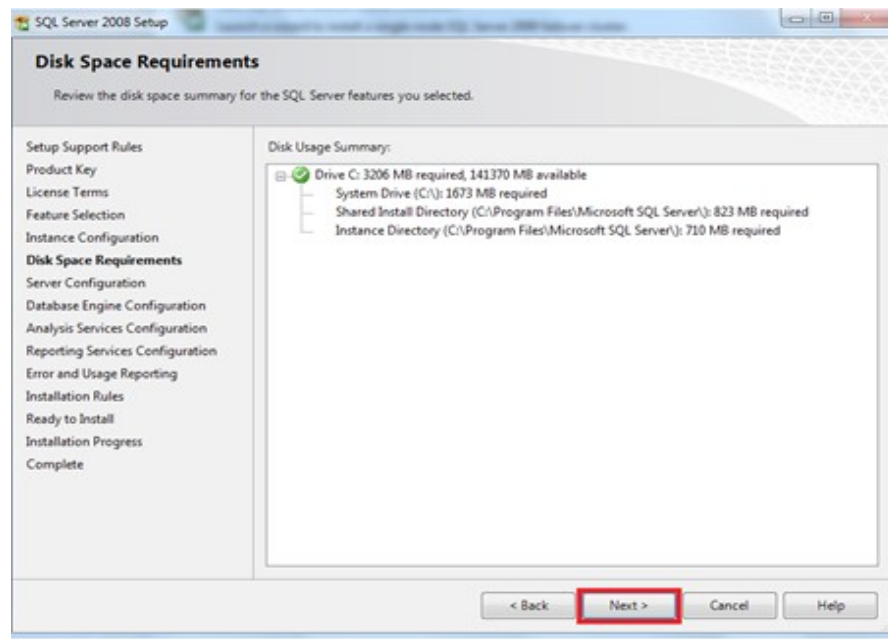




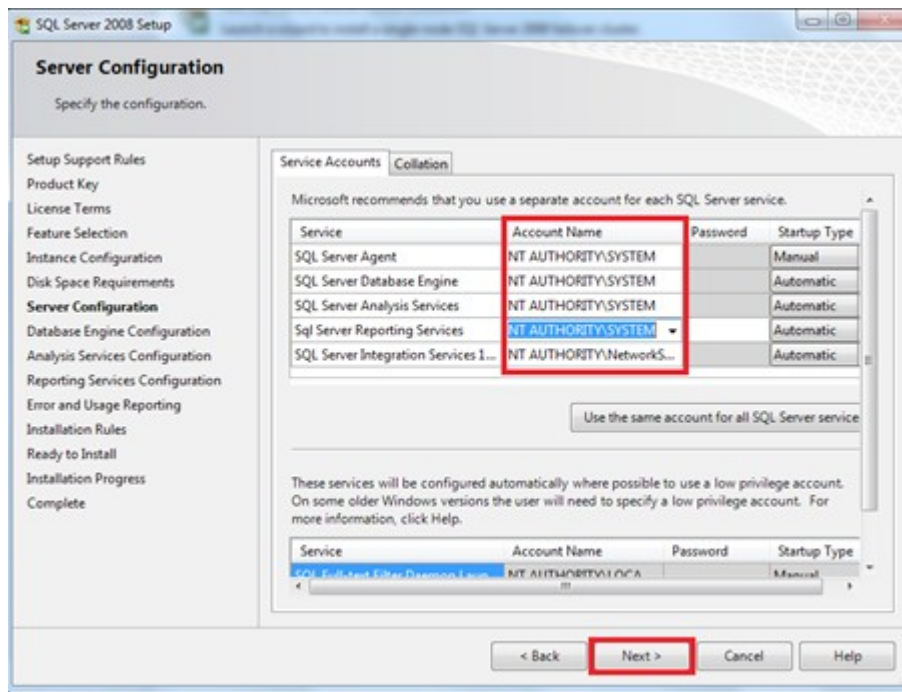
- La pantalla a continuación nos muestra la instancia que queremos instalar y uso la instancia “por defecto” y luego desde el Manager la cambio a mis necesidades, ahora clic en “siguiente”



- vemos que se nos presenta una pantalla en donde debemos verificar el espacio del disco, que debe ser suficiente para el trabajo del motor de base de datos, si no tenemos problema alguno seleccionamos “siguiente”

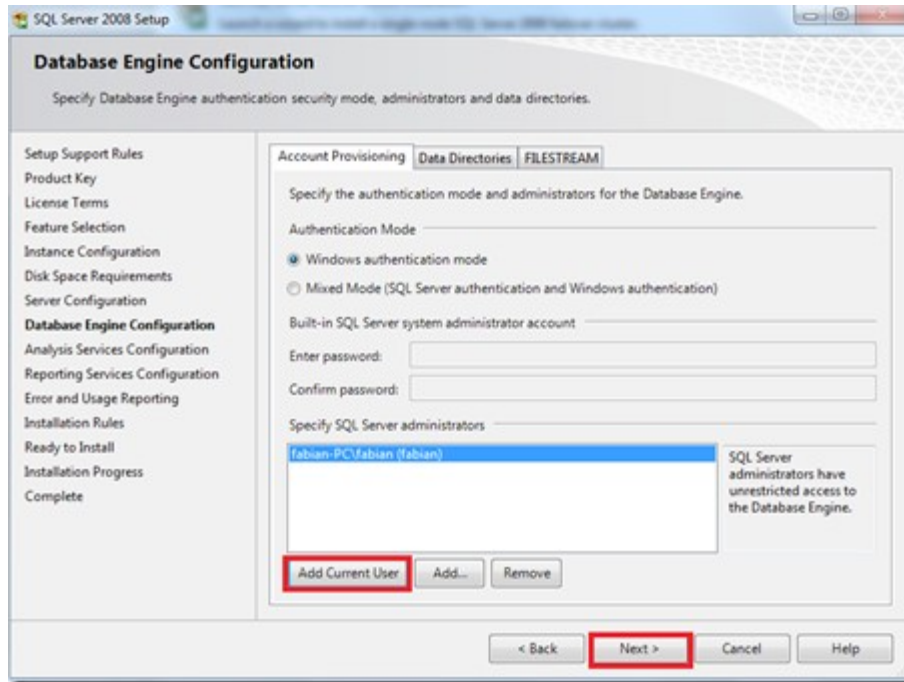


- selecciono “Autorización\Sistema” en todas las área de ingreso y clic en “Siguiete”

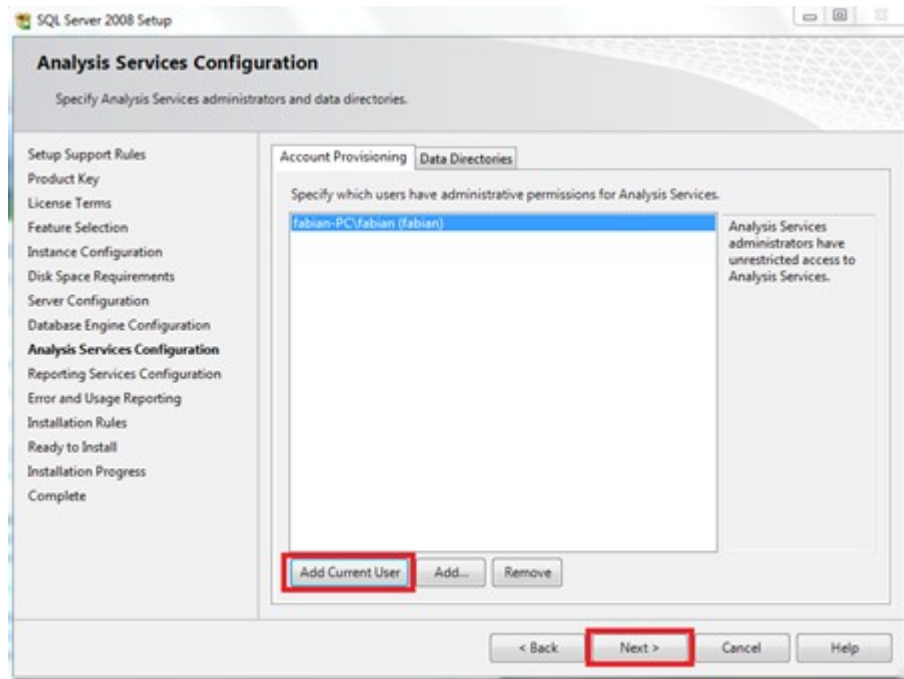




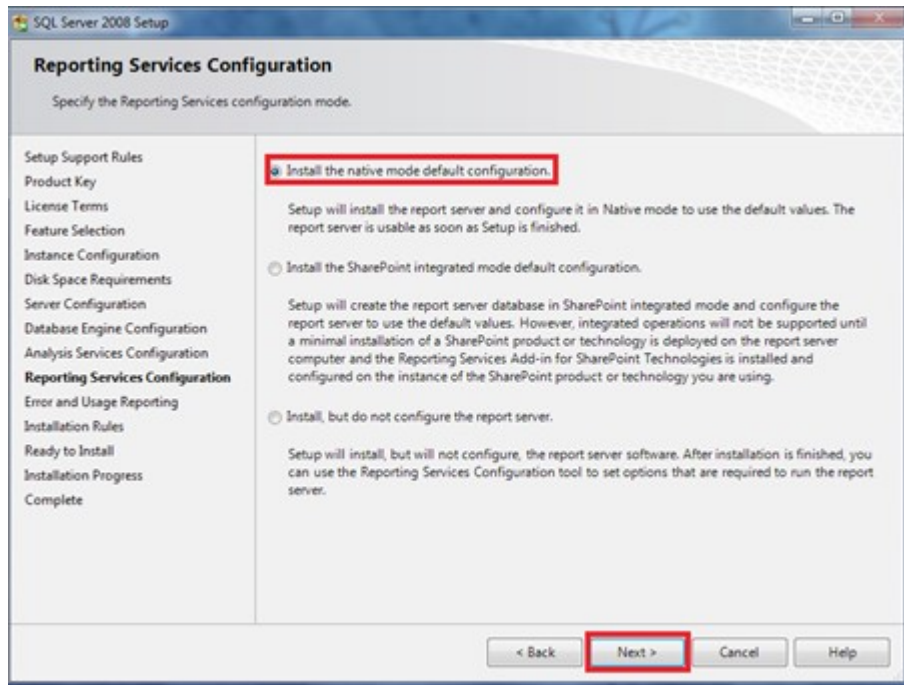
- la siguiente configuración tiene 2 opciones, yo elijo “Autenticación de Windows”, clic en “agregar un usuario existente” y “siguiente”



- Para el servicio de análisis debemos asignar un usuario, si queremos que el USER actual sea el mismo para el análisis de SQL seleccionamos “Agregar usuario existente” y “Siguiente”

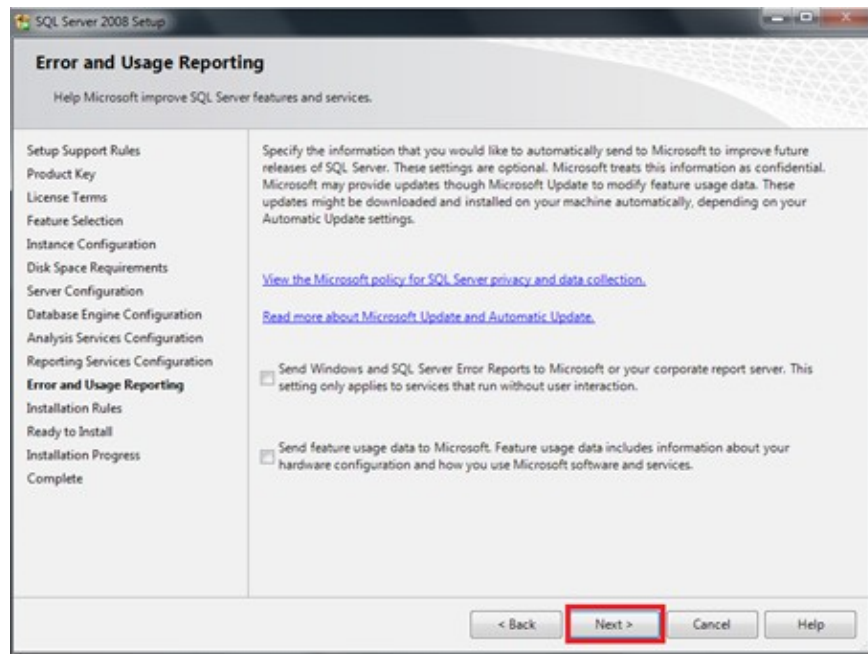


- Elegimos “configuración nativa” y “Siguiente”

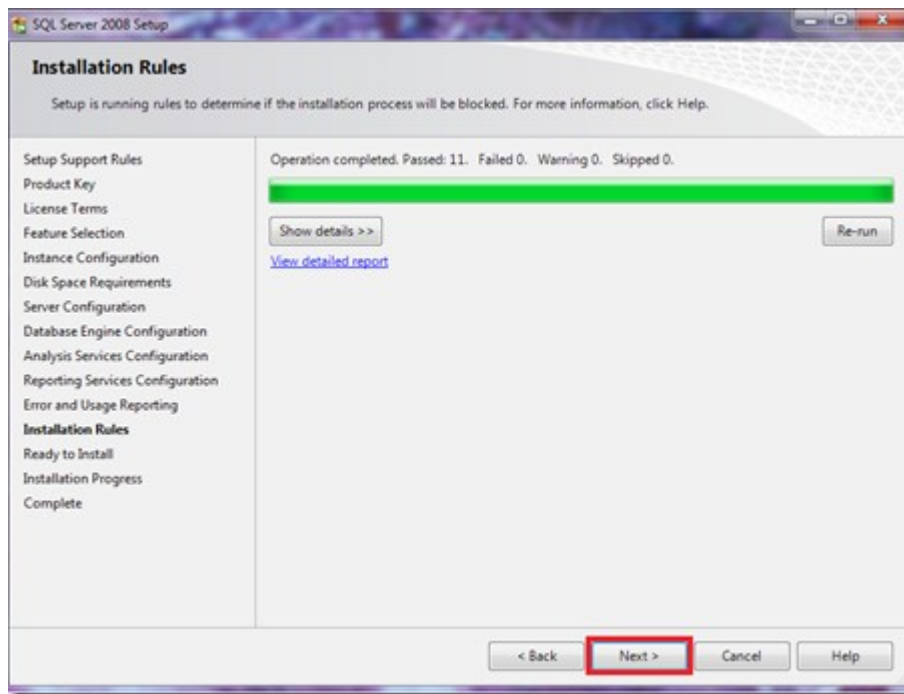




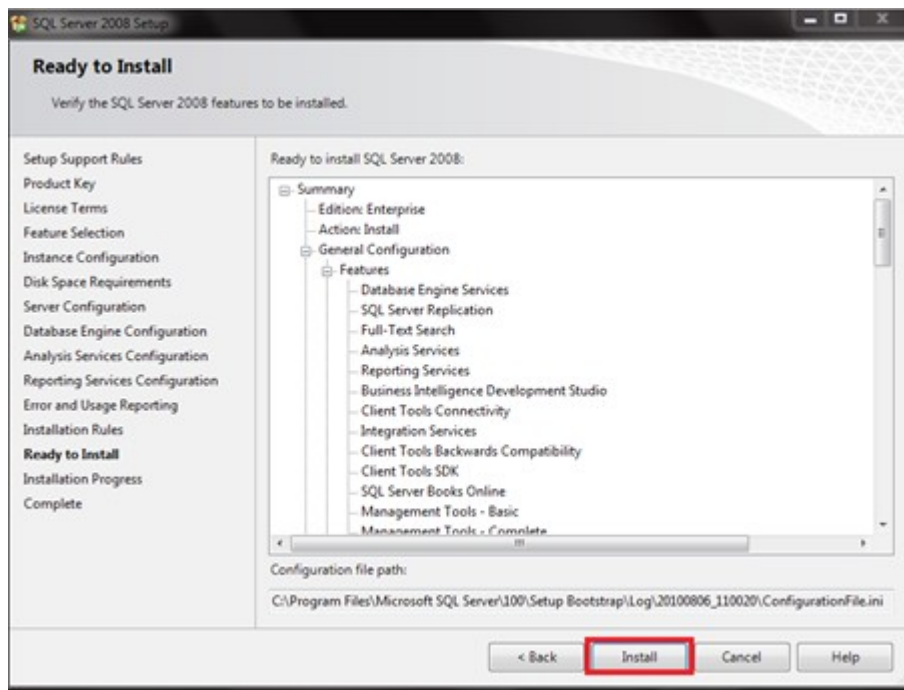
- En la siguiente ventana seleccionamos si deseamos enviar reportes de errores o características a Microsoft, si lo deseamos damos clic en cada checkbox o simplemente “siguiente”



- ahora instalemos todas las reglas de Instalación, que son en un número de 11 y “Siguiente”

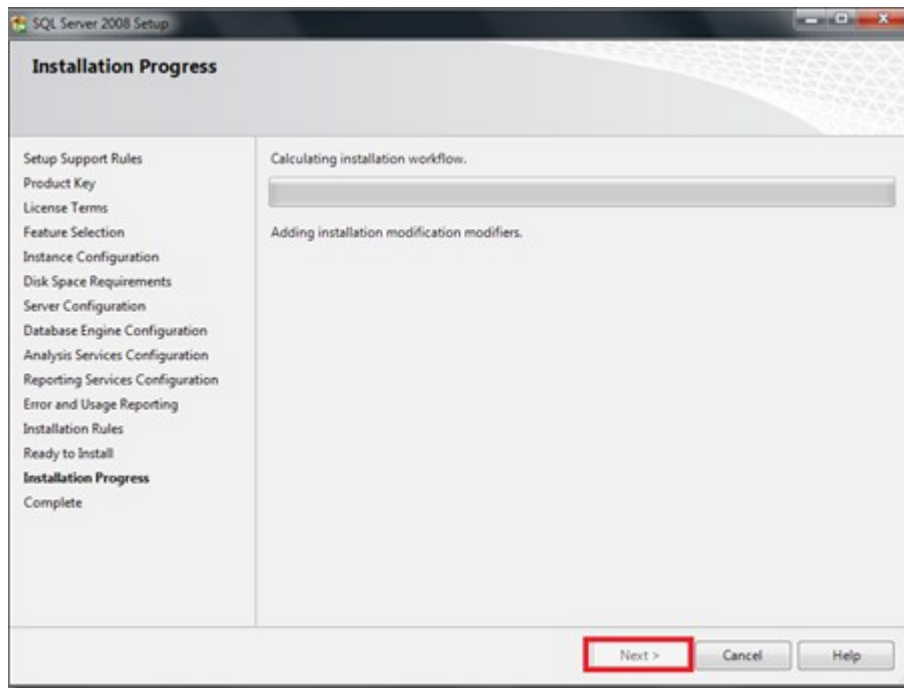


- Ahora si está todo listo para instalarse, vemos el resumen de los componentes que se agregarán, y si está todo bien clic en “Instalar”

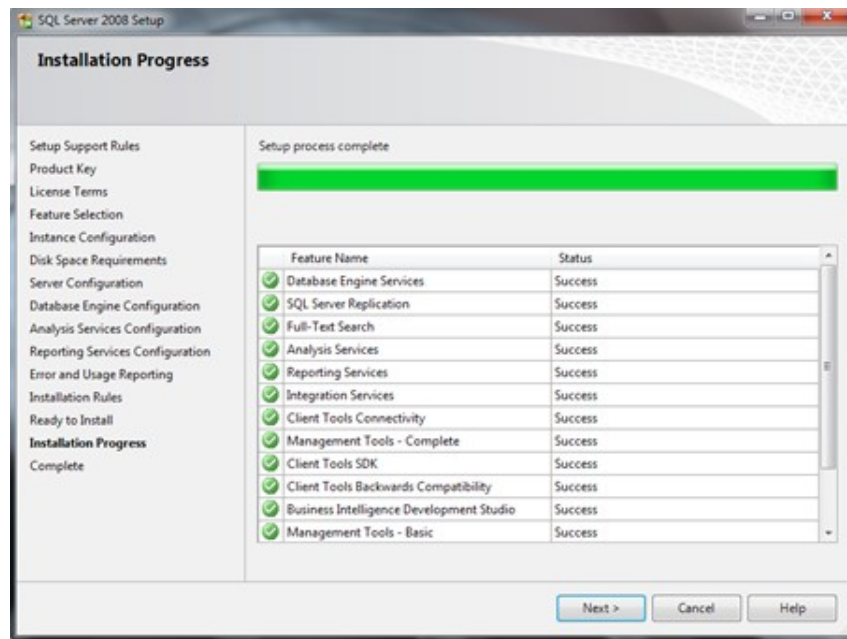




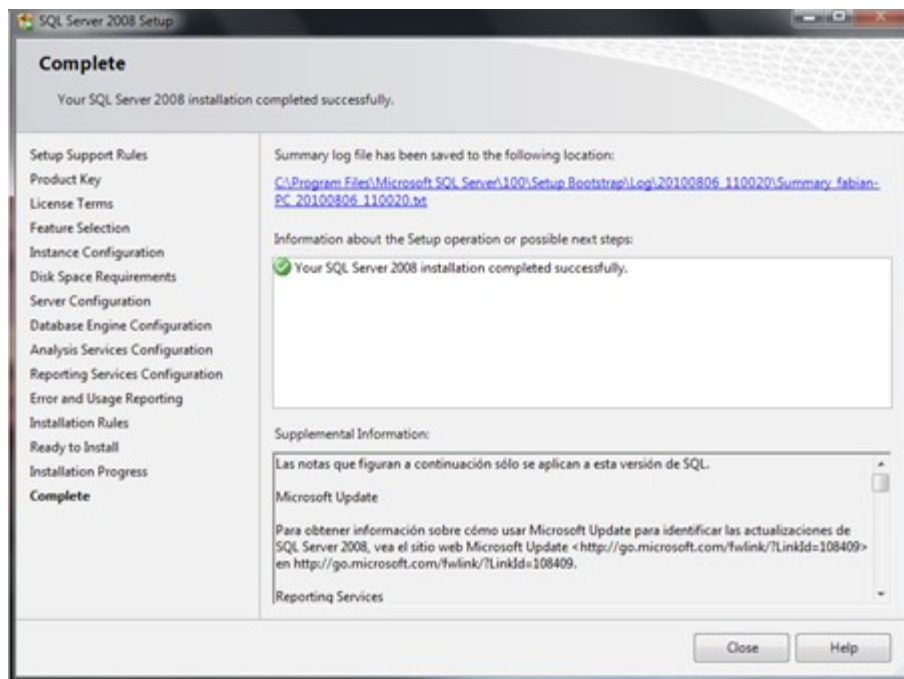
- Inicia nuestra instalación, debemos ser pacientes toma alrededor de 45 a 60 minutos, dependiendo de las características de Hardware del equipo en donde lo instalamos



- después de la espera se completa la instalación mostrándonos si existieron errores, en este caso todo se instaló con normalidad y seleccionamos “Siguiente”



- La pantalla final nos avisa que SQL Server 2008 está operando e instalado correctamente



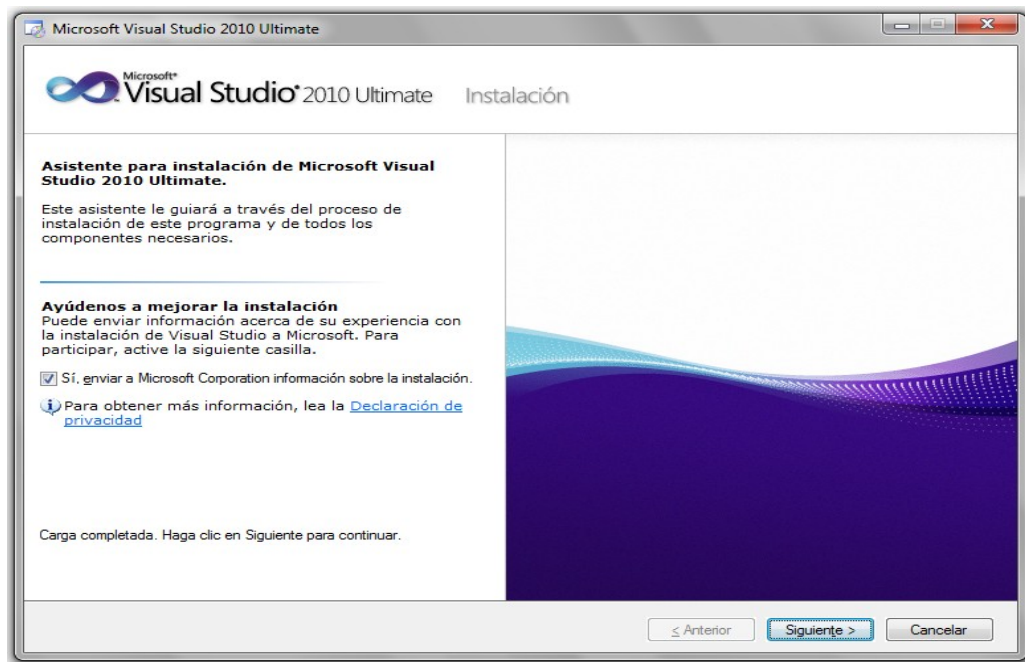


4.15.2 Instalación de Visual Estudio 2010

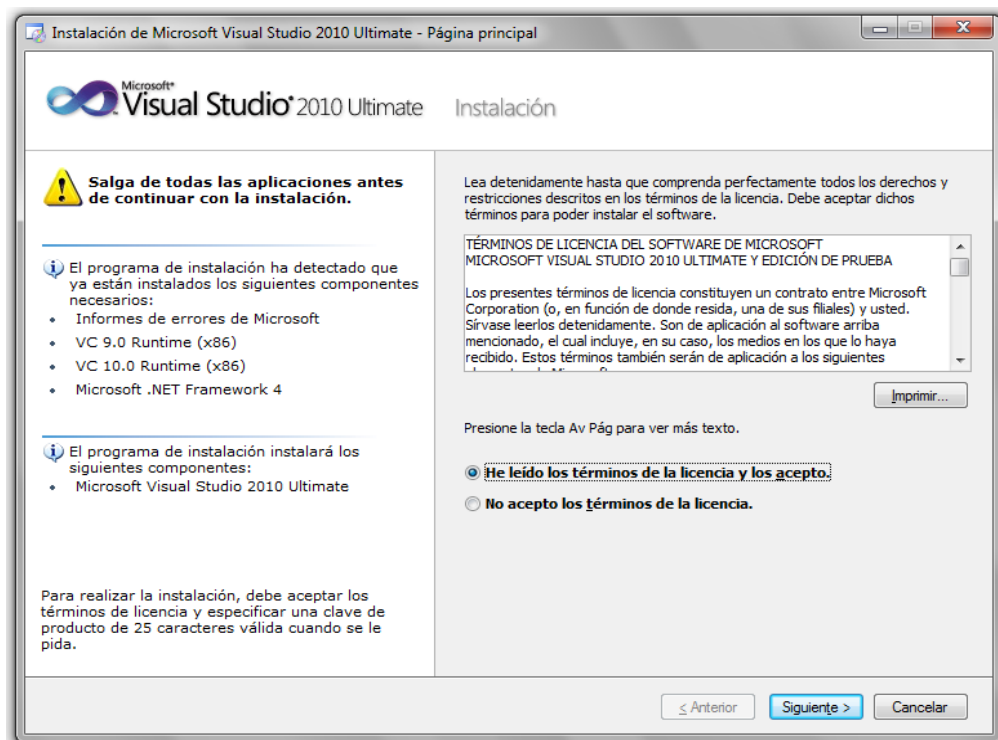
- Una vez que se ejecute el instalador de Visual Studio 2010, aparecerá una ventana de Instalación de Microsoft Visual Studio 2010, en la parte posterior damos click en la primera opción.



- Ejecuta el Asistente para la instalación, esperar hasta que abra todos los paquetes y luego hacer clic en Siguiente.

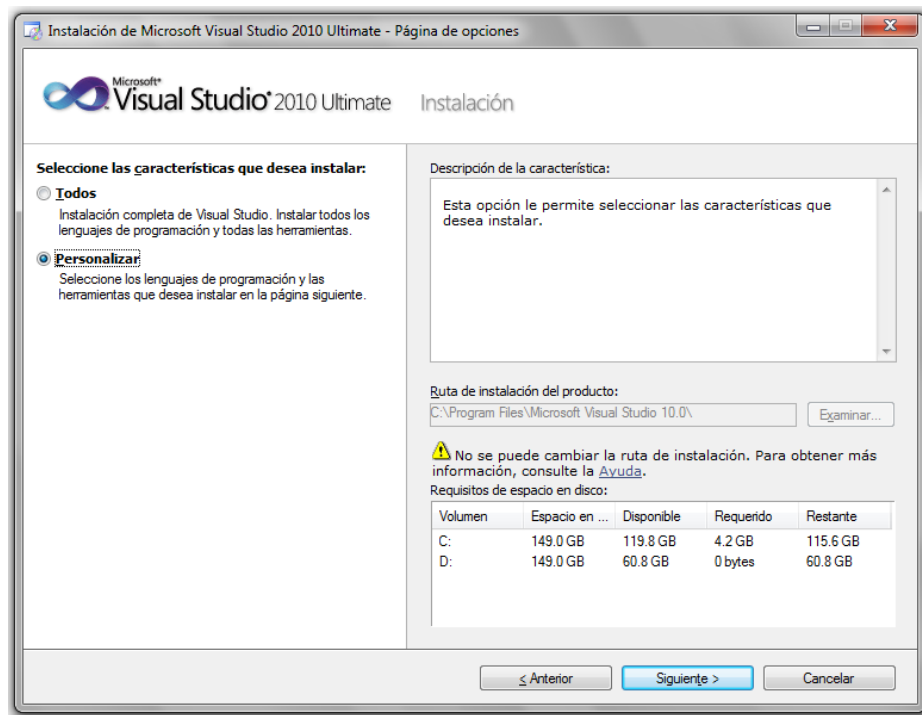


- En la ventana License Terms, clic He leído los términos de licencia y los acepto y clic en Siguiente.

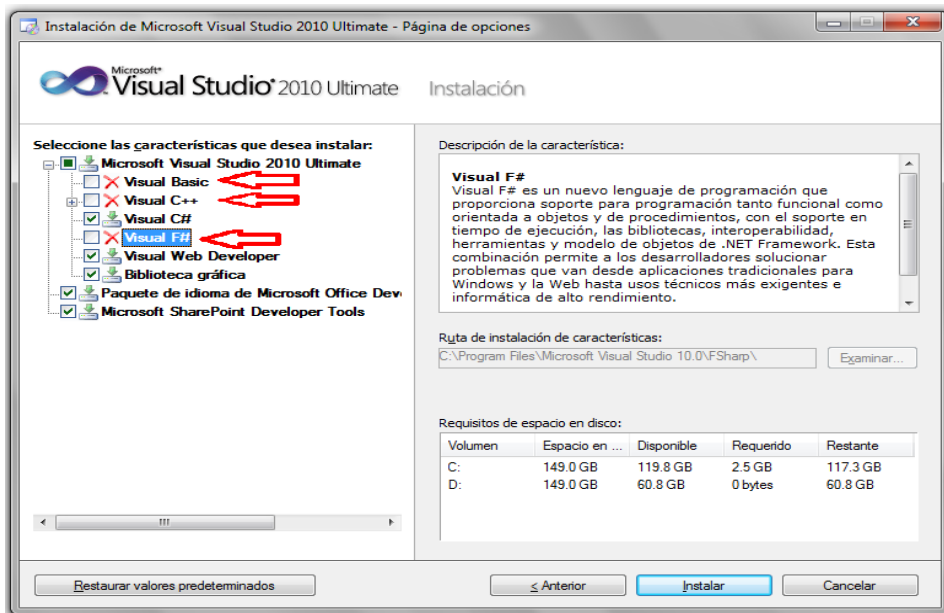




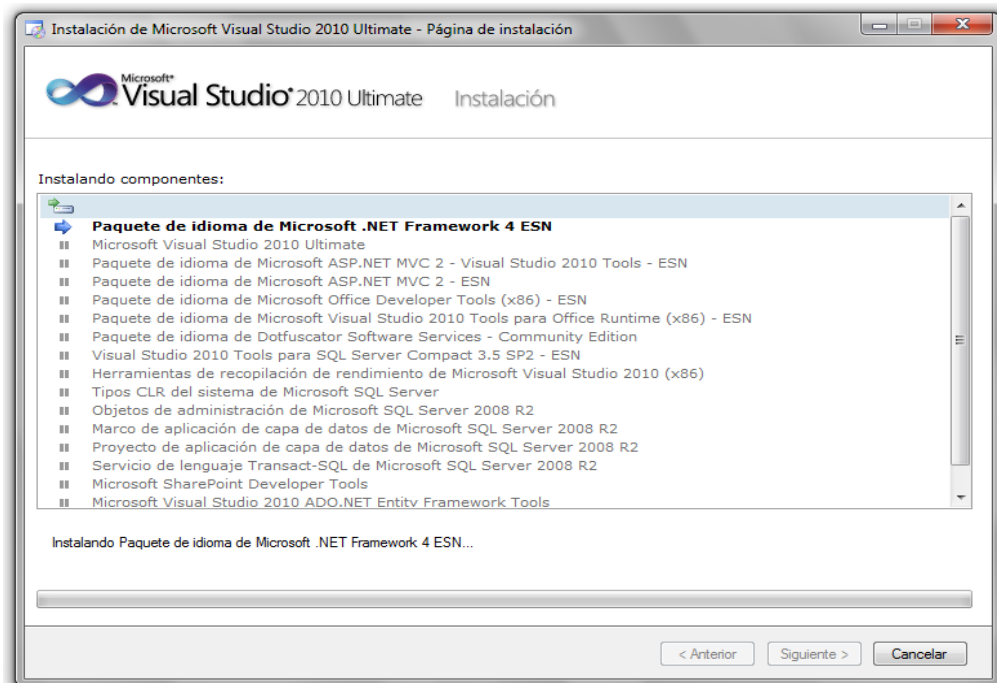
- En la ventana Página de Opciones valide que cumple con los requerimientos mínimos de espacio en disco, Clic en Personalizar y luego Clic en Siguiente.



- En vista de que en el anterior paso seleccionamos la opción de personalizar esta ventana de opciones nos indica las características que debemos instalar



- Se abre una ventana en donde se va a instalar los paquetes para el funcionamiento del programa. Este proceso durara dependiendo de la velocidad de procesamiento y de memoria, así que tenemos que esperar varios minutos.





4.16 Recopilación y Carga de Datos.

Con la finalidad de poder establecer los datos adecuados se realizó las pruebas necesarias, con lo que se validó la funcionalidad del sistema especialmente los reportes generados de la base de datos y al despliegue de resultados como consecuencia de los filtros de información que tiene que realizar a fin de poder visualizar.

Posteriormente se realizara una carga de datos considerable para demostrar la robustez del prototipo y la rapidez del mismo, igualmente las consultas tanto de los Usuarios, Clientes, Empresas y Retenes se lo realiza en la carga de datos inicial, lo que significa que a su vez se pueden realizar los mantenimientos necesarios en cualquiera de estas tablas.

4.17 Pruebas y Depuración Final en Funcionamiento

Una vez terminado el sistema realizaremos las pruebas y depuración final tanto de la base de datos, como del sistema, y a su vez probar las seguridades, más importantes para que así no sea jaqueado, para ello probamos primeramente la base de datos que esté correctamente instalada en el servidor de datos de la empresa, luego instalamos nuestro sistema en el servidor web, configurándolo, y verificando si está conectado a la base de datos del servidor de la empresa. Una vez que hemos realizado toda la instalación de la base de datos como también de nuestro sistema, procedemos a comprobar si hace falta corregir ciertos errores de compilación o configuración para así terminar con todas las pruebas.

4.18 Puesta en Marcha del Sistema

Una vez que se instala y configura la Base de Datos, se tiene que definir el funcionamiento del ambiente del mismo que almacenara los datos de los Usuarios, Clientes, Direcciones, Retenes y Empleados de la base son el Administrador los cuales tenemos que configurar y desbloquear, posteriormente se tendrá que realizar las interfaces de comunicación entre



la base de datos y a su vez, el funcionamiento en la WEB, para probar la comunicación de nuestro sistema a la base de datos de la empresa.

4.19 Capacitación al Usuario Final.

Para los usuarios del sistema, los capacitaremos mediante manuales que serán creados, para que así, ellos tengan la facilidad de manipular el sistema con respecto a su funcionamiento, claro que además de los manuales también tendrán la asesoría propia y personal, para evitar dificultades o problemas que se presenten.

4.19.1. Datos Informáticos

Empresa:	SERCOMEX
Número de Trabajadores de la Empresa:	20
Nombre Capacitador:	Xavier Ochoa
Material Entregable:	Manual de Usuario
Método de Capacitación:	Charla con Audiovisuales.
Tiempo de Capacitación:	20 horas.

4.19.2. Metodología de Desempeños Esperados

LO QUE EL USUARIO ALCANZARA HACER

Al término de la capacitación del Sistema Geográfico Georeferenciado, el usuario (s), estará en la capacidad de manejar eficientemente los siguientes puntos:

- Entender el funcionamiento del Sistema.
- Ejecutar el sistema con su respectiva clave de usuario.
- Lograr Ingresar clientes, con sus respectivos datos.
- Realizar la búsqueda de clientes.



- Visualizar los datos en los mapas.

LO QUE EL USUARIO DEBE SABER HACER

Para poder iniciar la ejecución del Sistema, el usuario previamente debe conocer y saber usar los siguientes tópicos:

- Windows y Office Básico.
- Matemática Básica.
- Conceptos Básicos de Informática.

4.19.3. Metodología de Diseño Operativo

ACTIVIDAD	RECURSOS	RESULTADOS
Charla Audiovisual: Revisión por parte del capacitador sobre el funcionamiento del Sistema.	Manual de Usuario, Diapositivas: Infocus,	Participar con argumentos relevantes. Resolver inquietudes de otros compañeros.
Laboratorio: Realizar practica de utilización del Sistema.	Computadoras Software	Vincular al usuario con el Sistema.
Prueba Diagnostica: Conocimientos adquiridos en la charla	Computadoras Software	Demostrar el conocimiento del Sistema

Diseño Operativo

Tabla # 16

Fuente: Propia del Autor

4.19.4. Observaciones y Recomendaciones

- El usuario deberá asistir a la charla revisando el Manual de Usuario antes entregado por parte de la Empresa.
- Participación analítica y crítica en la charla.

4.20 Capacitación al Personal Técnico.



Para la capacitación del personal técnico se creará del mismo modo manuales para explicar cómo está formado y creado el sistema, dentro de estos manuales existirá información, donde proporcione respuestas a problemas que se vayan generando durante su funcionamiento, en caso de que el sistema presente problemas de funcionamiento, además se les capacitará personalmente indicando como se puede solucionar problemas sencillos de tal forma que sean capaces de solucionar de una forma inmediata alguna dificultad que se presente.

4.20.1. Datos Informáticos

Empresa:	SERCOMEX
Número de Trabajadores de la Empresa:	3
Nombre Capacitador:	Paúl Vallejo.
Material Entregable:	Manual Técnico del Sistema.
Método de Capacitación:	Charla con Audiovisuales.
Tiempo de Capacitación:	10 horas.

4.20.2. Metodología de Desempeños Esperados

LO QUE EL PERSONAL TÉCNICO ALCANZARÁ HACER

Al término de la capacitación del Sistema Geográfico Georeferenciado, el técnico (s) de la Empresa, estará en la capacidad de manejar eficientemente los siguientes puntos:

- Entender el funcionamiento del Sistema.
- Ejecutar el sistema con su respectiva clave de administrador.
- Lograr Ingresar al mantenimiento de clientes
- Realizar la respectiva auditoria del Sistema
- Monitorear la Base de Datos.

LO QUE EL PERSONAL TÉCNICO DEBE SABER HACER



Para poder iniciar la ejecución del Sistema, el personal técnico previamente debe conocer y saber usar los siguientes tópicos:

- Manejo de Base de Datos
- Programación en Visual Studio 2010.
- SQL Server 2008.
- Manual de Procesos y Procedimientos
- Auditoría de Sistemas.

4.20.3. Metodología de Diseño Operativo

ACTIVIDAD	RECURSOS	RESULTADOS
Charla Audiovisual: Revisión del funcionamiento del Sistema, Base de Datos, Metodología de Desarrollo, Plataforma de Desarrollo.	Manual Técnico, Diapositivas: Mapas Conceptuales,	Participar con argumentos relevantes. Resolver inquietudes de otros compañeros.
Laboratorio: Demostración del Sistema	Laboratorio Computadoras Software	Vincular al personal técnico con el Sistema.

Diseño Operativo

Tabla # 17

Fuente: Propia del Autor

4.20.4. Observaciones y Recomendaciones

- El personal técnico deberá asistir a la charla revisando el Manual Técnico antes entregado por parte de la Empresa.

Participación analítica y crítica en la charla.



CAPÍTULO V

PRINCIPALES IMPACTOS

5.1 Científico.

La investigación desarrollada por el proyecto tiene una característica especial ya que aplicativos y tecnologías implementadas en nuestro medio únicamente se lo realizaba anteriormente con un software en dos capas, lo que implica que el soporte técnico necesario, es mucho más difícil, por lo tanto al tratarse de una implementación web se podrá beneficiar los técnicos y los demás usuarios, que utilizaran nuestro sistema, ya que cualquier inquietud y soporte se lo podrá arreglar sin para las reglas del negocio.

5.2 Educativo.

El desarrollo de un sistema informático involucra el conocimiento del diseño y la programación, por lo tanto al realizar un proyecto de automatización de procesos, se realizara un aporte muy importante para el conocimiento de los alumnos de la institución que se encuentran en los niveles inferiores los cuales se pueden motivar y desarrollar proyectos con mayor complejidad y de fusión de muchas tecnologías existentes.

5.3 Técnico.

Lo referente a este tema se considera las herramientas que se utilizaron para el modelamiento y desarrollo del SISTEMA GEOGRAFICO GEOREFERENCIADO, considerando que las mismas fueron estudiadas en las aulas de la Institución, en consecuencia su aplicación no tuvo mayor dificultad.



5.4 Tecnológico.

El ejecutar temas de innovación tecnológica aplicando los conocimientos recibidos en las aulas hace que cualquier proyecto sea considerado como un aporte institucional hacia la sociedad, mejor aun si lo que se resta realizando es la unión de varias tecnologías en un solo aplicativo, que es producto de la investigación tecnológica.

5.5 Empresarial.

El desarrollo de este tipo de sistemas nos da la posibilidad de abrirnos campo en el mercado haciéndonos conocer por este sistema, debido a que ya es algo tangible y funcional de alta calidad que está enfocado a perdurar en producción por un periodo de tiempo considerable, justificando la inversión de la empresa auspiciante.

5.6 Social.

La implementación de este sistema y la estructuración del mismo en la empresa, lograra crear fuentes de trabajo tanto en el ámbito administrativo como técnico, por consiguiente se estará brindando un aporte al ámbito social.

5.7 Económico.

El proyecto a implementarse no tendrá costo alguno para la empresa, por tal motivo es de gran interés contar con una infraestructura tecnológica que agilite los procesos y que sea muy eficiente y confiable sin pérdida de tiempo y dinero.

5.8 Conclusiones.

1. La metodología de investigación científica aplicada al proyecto permitirá determinar la utilización de tipos de investigación, métodos y herramientas que facilitaron la obtención de resultados cuantitativos y cualitativos, que



proporcionaron la obtención de procedimientos, tareas y procesos propios de la toma de decisiones del Sistema GEOGRÁFICO GEOREFERENCIADO.

2. La aplicabilidad de los pasos que se siguen en ingeniería de software fueron determinantes ya que mediante el análisis permitieron discernir todo el flujo de información que se ejecuta en los procesos, al mismo tiempo poder diseñar adecuadamente los procesos relaciones, las bases de datos y sus objetos, a fin de poder obtener una organización metódica y bien estructurada de nuestro sistema.
3. Los conocimientos adquiridos en el Instituto tecnológico Superior Cordillera, me permitieron consolidar las teorías pragmáticas en herramientas de ejecución prácticas utilizando lenguajes de programación, y bases de datos materializadas en conocimientos de programación informática. Lo que definió en una concatenación de ideas lógicamente estructuradas y enlazadas hacia la consecución de un objetivo general.
4. En un entorno de desarrollo de software se debe realizar muchas pruebas de campo como validaciones de sistema, validaciones de usuarios, para poder decir que el software está terminado y en funcionamiento, más aun si consideramos que en nuestro caso también se tiene que mantener un especial atención por la fusión de tecnologías ejecutadas en el proyecto, lo que permitirá la consolidación de la información estructural y organizada para la utilización de cada usuario según su rol que desempeña.



5. Hay que mencionar el hecho que para el desarrollo de este tipo de tecnologías es indispensable utilizar lenguajes de tercera generación y orientada a objetos, para poder lograr un alto rendimiento en el manejo de este tipo de ambientes tecnológicos.

6. Con este tipo de sistemas integrados me pude dar cuenta que como alumno de la carrera de sistemas no solo debo instruirme en mi especialidad sino también interactuar con otras, como por ejemplo el manejo y conocimiento de este lenguaje de programación Web Asp .Net que es tema fundamental dentro de mi desarrollo, ya que por medio de este conocimiento logramos realizar nuestro sistema en la Web.

5.9 Recomendaciones.

1. Todo trabajo de investigación debe sustentarse con una metodología métodos y herramientas de la investigación científica por lo tanto es recomendable que las personas se preparen en este campo para realizar cualquier investigación, mas aun considerando que esto es la base para poder realizar el proyecto de grado previo a la obtención del título de Tecnólogo Analista de Sistemas.



2. Seguir la misma secuencia de desarrollo de software a fin de poder llegar a obtener un sistema completo, aplicable y amigable, es el resultado de haber seguido una metodología de desarrollo y técnicas de programación que permita interactuar los datos alfanuméricos (información), con los datos y relacionarlos, a fin de poder estructurar un sistema real.
3. Para cualquier tipo de integración de tecnologías avanzadas es indispensable conocer los diversos lenguajes de programación por lo cual es recomendable instruir a los alumnos en la nuevas herramientas de desarrollo de software que cada día se van actualizando y que nos ofrecen mayores facilidades para integrar sistemas de muy alto rendimiento.
4. Siempre será importante conocer que en el mundo del desarrollo de software existen diversos tipos de caminos o integraciones con otros sistemas y especialidades, por lo que se debería agregar como materia principal el manejo de diversas tecnologías, ya que en el mundo laboral el tecnólogo debe estar listo para programar o resolver cualquier tipo de problema.
5. Como otra recomendación muy importante podemos destacar que dentro de las seguridades de la zona desmilitarizada, la empresa solamente cuenta con un firewall, que lo convierte en una red muy vulnerable al hackeo.
6. Como última recomendación que les damos es que utilicen el Navegador Firefox (Cualquier versión), Ya que este navegador respeta todos los estándares WEB.



CAPITULO VI

BIBLIOGRAFÍA

6.1 Bibliografía

1. Burton Harvey, Simon Robinson, Julian Templeman y Karli Watson , (2000), “C# Programming with the Public Beta”,Wrox Press
2. Simon Robinson, Burt Harvey, Craig McQueen, Christian Nagel, Morgan Skinner, Jay Glynn, Karli Watson, Ollie Cornes, JerodMoemeka,(2001),“Professional C#”, Wrox Press .
3. JesseLiberty,(2001),“Programming C#”, O’Reilly
4. Eric Gunnerson(200),“A programmer’s introduction to C#”, Apress.
5. Jorge Serrano (2009) Manual Avanzado de Visual Basic 2008, editorial Anaya Multimedia, edición 2.
6. Jorge Serrano (2010) Manual Avanzado de Visual Basic 2010, editorial Anaya Multimedia, edición 1.



9 6.2 Net Grafía

1. Guía de referencia del lenguaje C#:
(<http://www.msdn.microsoft.com/net/ecma>)
2. Compilador de C# (incluido en el .NET SDK):
(<http://www.msdn.microsoft.com/code/sample.asp?url=/msdn-files/027/000/976/msdncompositedoc.xml>)
3. Versión beta de Visual Studio .NET:
(<http://msdn.microsoft.com/vstudio/nextgen/beta.asp>)
4. Portal de C# en español:
(<http://tdg.lsi.us.es/~csharp>)
5. Web de Microsoft:
www.msdn.microsoft.com
www.msdn.microsoft.com/net

6.3 GLOSARIO DE TÉRMINOS TÉCNICOS

ANSI: American National Standards Institute - Instituto Nacional de Normas de Estados Unidos.

Aplicación: Cualquier programa que corra en un sistema operativo y que haga una función específica para un usuario. Por ejemplo, procesadores de palabras, bases de datos, agendas electrónicas, etc.

DDL: Lenguaje de definición de datos (Sistemas gestores de bases de datos).



GNU: Las letras GNU se refieren en inglés a No es UNIX (Not UNIX); es un sistema de programas compatible con UNIX, desarrollado por el Free Software Foundation (FSF). El proyecto GNU empezó en 1983 por Richard Stallman del Massachusetts Institute of Technology. La filosofía detrás de GNU es producir programas que no sean propietarios. Cualquiera puede descargar, modificar y redistribuir programas GNU. La única restricción es que no le pueden limitar la redistribución. Linux utiliza muchos programas GNU, y en el pasado los sistemas GNU usaban el kernel de Linux. Esta estrecha relación entre Linux y GNU hace que muchas personas piensen, erradamente, que son lo mismo. Pero son bastante distintos, y GNU está desarrollando su propio kernel llamado HURD para reemplazar el kernel de Linux en los sistemas GNU.

GPL: Acrónimo en inglés de General Public License (Licencia Pública General). Esta licencia regula los derechos de autor de los programas de software libre (free software) promovido por el Free Software Foundation (FSF) en el marco de la iniciativa GNU. Permite la distribución de copias de programas (e incluso cobrar por ello), así como modificar el código fuente de los mismos o utilizarlo en otros programas.

IIS: Microsoft Internet Information Services. Servicios de Información de Internet de Microsoft. IIS es un conjunto de servicios basados en Internet, para máquinas con Windows. Compite con Apache en el área de servidores web.

INTERFAZ: Una conexión e interacción entre hardware, software y usuario, es decir como la plataforma o medio de comunicación entre usuario o programa.

MySQL: MySQL es uno de los Sistemas Gestores de Bases de Datos más populares. Su ingeniosa arquitectura lo hace extremadamente rápido y fácil de personalizar. La extensiva reutilización del código dentro del software y una aproximación minimalística para producir características altamente funcionales, ha dado lugar a un sistema de



administración de base de datos de alta velocidad, compactación, estabilidad y facilidad de despliegue. La exclusiva separación del core server del manejador de tablas, permite funcionar a MySQL bajo control estricto de transacciones o con acceso a disco no transaccional ultrarrápido. Lo mantiene la empresa sueca MySQL AB bajo la licencia GPL (GNU PublicLicense); puede utilizarse gratuitamente y su código fuente está disponible.

USUARIO: Cualquier individuo que interactúa con la computadora a nivel de aplicación. Los programadores, operadores y otro personal técnico no son considerados usuarios cuando trabajan con la computadora a nivel profesional.

ODBC: Open DatabaseConnectivity. Estándar de acceso a Bases de Datos desarrollado por Microsoft cuyo objetivo es hacer posible el acceder a cualquier dato de cualquier aplicación, sin importar qué Sistema Gestor de Bases de Datos (DBMS por sus siglas en inglés) almacene los datos, ODBC logra esto al insertar una capa intermedia llamada manejador de Bases de Datos, entre la aplicación y el DBMS.

Perl: PracticalExtraction and ReportLanguage -- Lenguaje de programación muy utilizado para la elaboración de aplicaciones CGI, principalmente para realizar consultas a bases de datos como Oracle, SQL-Server, SyBase, etc, o a herramientas locales como WAIS. Perl es un lenguaje para manipular textos, archivos y procesos, proporciona una forma fácil y legible para realizar trabajos que normalmente se realizarían en C o en un shell. Perl nació y se ha difundido bajo el sistema operativo UNIX, aunque existe para otras plataformas. Perl fue desarrollado por Larry Wall, y está distribuido libremente bajo la filosofía GNU.

Página Web: Resultado en hipertexto o hipermedia que proporciona un navegador del WWW después de obtener la información solicitada. Su contenido puede ir desde un texto corto a un voluminoso conjunto de textos, gráficos estáticos o en movimiento, sonido, etc. Algunas veces el citado término es utilizado incorrectamente en orden de



designar el contenido global de un sitio web, cuando en ese caso debería decirse "Web site".

Windows Communication Foundation o WCF (también conocido como Indigo): Es la nueva plataforma de mensajería que forma parte de la API de la Plataforma .NET 3.0 (antes conocida como WinFX, y que no son más que extensiones para la versión 2.0). Se encuentra basada en la Plataforma .NET 2.0 y de forma predeterminada se incluye en el Sistema Operativo Microsoft Windows Vista.

Windows PresentationFoundation (WPF): Es una tecnología de Microsoft, presentada como parte de Windows Vista. Permite el desarrollo de interfaces de interacción en Windows tomando las mejores características de las aplicaciones Windows y de las aplicaciones web.



ANEXOS

MANUAL DE USUARIO

INGRESO AL SISTEMA



Para ingresar el sistema se da doble click en el escritorio en el icono que representa a los navegadores en este caso Explorer



Al ingresar al sistema se presenta la siguiente pantalla:

LOGIN

Esta pantalla es la de inicio del sistema, aquí nos pedirá ingresar el nombre de usuario y la contraseña, si son correctos presenta el menú del sistema, caso contrario no nos permitirá ingresar.

INICIAR SESIÓN

Especifique su nombre de usuario y contraseña.

Información de cuenta

Nombre de usuario:

Contraseña:

☐ Mantenerme conectado

124



Iniciar sesión

Una vez ingresado los datos en sus respectivos campos presionamos **Iniciar sesión** si el password es incorrecto se muestra el siguiente mensaje

El intento de conexión no fue correcto. Inténtelo de nuevo.

Información de cuenta

Nombre de usuario:

Contraseña:

☐ Mantenerme conectado

Iniciar sesión

MANTENIMIENTOS



Una vez dentro, el sistema muestra la pantalla principal del sistema, esta nos presentará el siguiente formulario con la lista de usuarios, donde podemos eliminar y modificar los registros.

USUARIOS

[+ NUEVO](#)

LISTA DE USUARIOS

BUSCAR POR:

		CEDULA	NOMBRE	PRIMER APELLIDO	SEGUNDO APELLIDO
		1234567890	Juan	Gallegos	Lara

El siguiente Formulario nos muestra como ingresar datos del Usuario además de la opción de guardar los datos o crear uno nuevo

USUARIOS

[+ NUEVO](#) [GUARDAR](#)

NOMBRE:

PRIMER APELLIDO:

SEGUNDO APELLIDO:

CEDULA:

TELEFONO:

DEPARTAMENTO:

CARGO:

ROL:

TIPO USUARIO:

LISTA DE USUARIOS

El siguiente Formulario nos muestra como Modificar datos del Usuario además de la opción de guardar los datos modificados.



INSTITUTO TECNOLÓGICO SUPERIOR CORDILLERA

USUARIOS

[+ NUEVO](#) [GUARDAR](#)

NOMBRE:	<input type="text" value="Carlos"/>	SEGUNDO APELLIDO	<input type="text" value="Cedeño"/>
PRIMER APELLIDO	<input type="text" value="Aguilar"/>	TELEFONO	<input type="text" value="2428996"/>
CEDULA	<input type="text" value="1724596834"/>	CARGO	<input type="text" value="Jefe Sistemas"/>
DEPARTAMENTO	<input type="text" value="Sistemas"/>	TIPO USUARIO	<input type="text" value="Administrador"/>
ROL	<input type="text" value="SubJefe"/>		

[LISTA DE USUARIOS](#)

BUSQUEDA

En este formulario podemos buscar al cliente de la empresa y los retenes de policía cercanos respectivamente para ello podrán hacerlo por dirección o nombre del cliente.

Tipo de búsqueda:

☒ Todos

Nombre:

Nombre de usuario:

Password:



MANUAL DE TÉCNICO

SCRIPT BASE DE DATOS

Create table [Tbl_Coordenadas]

```
(  
    [Coor_Id] Integer NOT NULL,  
    [Coor_Longitud] Decimal(10,4) NULL,  
    [Coor_Latitud] Decimal(10,4) NULL,  
    [Coor_FechaCreacion] Datetime NULL,  
    [Coor_Estado] Char(1) NULL,  
    [Ret_Id] Integer NOT NULL,  
    [Cli_Id] Integer NOT NULL,
```

Primary Key ([Coor_Id])

```
)
```

go

Create table [Tbl_Clientes]



```
(
    [Cli_Id] Integer NOT NULL,
    [Cli_PrimerNombre] Varchar(20) NULL,
    [Cli_SegundoNombre] Varchar(20) NULL,
    [Cli_PrimerApellido] Varchar(20) NULL,
    [Cli_SegundoApellido] Varchar(20) NULL,
    [Cli_EstadoCivil] Varchar(10) NULL,
    [Cli_Ruc] Varchar(15) NULL,
    [Cli_Cedula] Varchar(10) NULL,
    [Cli_FechaCreacion] Datetime NULL,
    [Cli_Estado] Char(1) NULL,
    [TipUsu_Id] Integer NOT NULL,
    [Dir_Id] Integer NOT NULL,
    [Par_Id] Integer NOT NULL,
    Primary Key ([Cli_Id])
)
go
```

```
Create table [Tbl_Retenes]
(
    [Ret_Id] Integer NOT NULL,
    [Ret_Descripcion] Varchar(20) NULL,
    [Ret_FechaCreacion] Datetime NULL,
    [Ret_Estado] Char(1) NULL,
    Primary Key ([Ret_Id])
)
go
```



Create table [Tbl_Usuario]

```
(  
    [Usu_Id] Integer NOT NULL,  
    [Usu_Nombre] Varchar(20) NULL,  
    [Usu_PrimerApellido] Varchar(20) NULL,  
    [Usu_SegundoApellido] Varchar(20) NULL,  
    [Usu_Cedula] Varchar(10) NULL,  
    [Usu_Telefono] Varchar(10) NULL,  
    [Usu_login] Varchar(20) NULL,  
    [Usu_Pass] Varchar(10) NULL,  
    [Usu_Estado] Char(1) NULL,  
    [Dep_Id] Integer NOT NULL,  
    [Car_Id] Integer NOT NULL,  
    [Rol_Id] Integer NOT NULL,  
    [TipUsu_Id] Integer NOT NULL,
```

Primary Key ([Usu_Id])

```
)  
go
```

Create table [Tbl_Rol]

```
(  
    [Rol_Id] Integer NOT NULL,  
    [Rol_Descripcion] Varchar(20) NULL,  
    [Rol_Estado] Char(1) NULL,
```

Primary Key ([Rol_Id])

```
)  
go
```



Create table [Tbl_TipoUsuario]

```
(  
    [TipUsu_Id] Integer NOT NULL,  
    [TipoUsu_Descripcion] Varchar(20) NULL,  
    [TipoUsu_FechaCreacion] Datetime NULL,  
    [TipoUsu_Estado] Char(1) NULL,  
Primary Key ([TipUsu_Id])  
)  
go
```

Create table [Tbl_Telefonos]

```
(  
    [Tel_Id] Integer NOT NULL,  
    [Tel_Numero] Varchar(10) NULL,  
    [Tel_Estado] Char(1) NULL,  
    [Cli_Id] Integer NOT NULL,  
Primary Key ([Tel_Id])  
)  
go
```

Create table [Tbl_Direcciones]

```
(  
    [Dir_Id] Integer NOT NULL,  
    [Dir_AvPrincipal] Varchar(20) NULL,  
    [Dir_AvSecundaria] Varchar(20) NULL,  
    [Dir_NumCasa] Varchar(10) NULL,  
    [Dir_FechaCreacion] Datetime NULL,  
    [Dir_Estado] Char(1) NULL,  
Primary Key ([Dir_Id])  
)  
go
```



Primary Key ([Dir_Id])

)

go

Create table [Tbl_Departamento]

(

[Dep_Id] Integer NOT NULL,

[Dep_Descripcion] Varchar(50) NULL,

[Dep_Estado] Char(1) NULL,

Primary Key ([Dep_Id])

)

go

Create table [Tbl_Cargo]

(

[Car_Id] Integer NOT NULL,

[Car_Descripcion] Varchar(20) NULL,

[Car_Estado] Char(1) NULL,

Primary Key ([Car_Id])

)

go

Create table [Tbl_Parametros]

(

[Par_Id] Integer NOT NULL,

[Par_CodigoPadre] Integer NULL,

[Par_CodigoHijo] Integer NULL,

[Par_Descripcion] Varchar(100) NULL,



```
[Par_Estado] Char(1) NULL,  
Primary Key ([Par_Id])  
)  
go
```

```
Alter table [Tbl_Coordenadas] add foreign key([Cli_Id]) references [Tbl_Clientes] ([Cli_Id])  
on update no action on delete no action  
go
```

```
Alter table [Tbl_Telefonos] add foreign key([Cli_Id]) references [Tbl_Clientes] ([Cli_Id]) on  
update no action on delete no action  
go
```

```
Alter table [Tbl_Coordenadas] add foreign key([Ret_Id]) references [Tbl_Retenes]  
([Ret_Id]) on update no action on delete no action  
go
```

```
Alter table [Tbl_Usuario] add foreign key([Rol_Id]) references [Tbl_Rol] ([Rol_Id]) on  
update no action on delete no action  
go
```

```
Alter table [Tbl_Clientes] add foreign key([TipUsu_Id]) references [Tbl_TipoUsuario]  
([TipUsu_Id]) on update no action on delete no action  
go
```

```
Alter table [Tbl_Usuario] add foreign key([TipUsu_Id]) references [Tbl_TipoUsuario]  
([TipUsu_Id]) on update no action on delete no action  
go
```

```
Alter table [Tbl_Clientes] add foreign key([Dir_Id]) references [Tbl_Direcciones] ([Dir_Id])  
on update no action on delete no action  
go
```



```
Alter table [Tbl_Usuario] add foreign key([Dep_Id]) references [Tbl_Departamento]
([Dep_Id]) on update no action on delete no action
go
Alter table [Tbl_Usuario] add foreign key([Car_Id]) references [Tbl_Cargo] ([Car_Id]) on
update no action on delete no action
go
Alter table [Tbl_Clientes] add foreign key([Par_Id]) references [Tbl_Parametros] ([Par_Id])
on update no action on delete no action
go
Set quoted_identifier on
go
Set quoted_identifier off
go
```

CÓDIGO FUENTE

USUARIO LOGICA

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.Linq;
using Acceso_Datos;

namespace Logica_Negocios
{
    public class Usuario_Logica
```



```
{  
    public static SistemaGeograficoDataContext dc = new  
SistemaGeograficoDataContext();  
    public static List<Tbl_Usuario> ObtenerUsuarios()  
    {  
        var usuario = dc.Tbl_Usuario.Select(usu => usu);  
        return usuario.ToList();  
    }  
  
    public static Tbl_Usuario ObtenerusuarioId(int id)  
    {  
        try  
        {  
            var usuario = dc.Tbl_Usuario.Single(usu => usu.Usu_Id == id);  
            return usuario;  
        }  
        catch (Exception)  
        {  
            return null;  
        }  
    }  
  
    private static void Validar(Tbl_Usuario Tbl_Usuario)  
    {  
        string mensaje = "";  
  
        if (Tbl_Usuario.Usu_Nombre == "")
```



```
{  
    mensaje += "INGRESE NOMBRE </br>";  
}  
  
if (mensaje != "")  
{  
    throw new ArgumentException(mensaje);  
}  
  
}  
  
public static void save(Tbl_Usuario Tbl_Usuario)  
{  
    Tbl_Usuario.Usu_Id = Secuencial();  
    Tbl_Usuario.Usu_Estado = 'A';  
  
    try  
    {  
        Validar(Tbl_Usuario);  
        dc.Tbl_Usuario.InsertOnSubmit(Tbl_Usuario);  
        dc.SubmitChanges();  
    }  
    catch (Exception ex)  
    {  
  
        throw new ArgumentException("DATOS NO INSERTADOS\n" + ex.Message);  
    }  
}
```




```
}  
}
```

```
public static void Modificar(Tbl_Usuario Tbl_Usuario)  
{  
    try  
    {  
        dc.SubmitChanges();  
    }  
    catch (Exception ex)  
    {  
  
        throw new ArgumentException("DATOS NO MODIFICADOS\n" + ex.Message);  
    }  
}  
  
public static void eliminar(Tbl_Usuario Tbl_Usuario)  
{  
    try  
    {  
        dc.SubmitChanges();  
    }  
    catch (Exception ex)  
    {  
        throw new ArgumentException("No se pudo Eliminar el datos \n" + ex.Message);  
    }  
}
```



```
public static int Secuencial()
{
    int codigo = 0;
    try
    {
        codigo = dc.Tbl_Usuario.Max(usu => usu.Usu_Id + 1);
        return codigo;
    }
    catch (Exception)
    {
        return 1;
    }
}

}
```

TIPO USUARIO LOGICA

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.Linq;
using Acceso_Datos;
```



namespace Logica_Negocios

```
{  
    public class TipoUsuario_Logica  
    {  
        public static SistemaGeograficoDataContext dc = new  
SistemaGeograficoDataContext();  
        public static List<Tbl_TipoUsuario> ObtenerTipoUsuario()  
        {  
            var usuario = dc.Tbl_TipoUsuario.Select(usu => usu);  
            return usuario.ToList();  
        }  
  
        public static Tbl_TipoUsuario ObtenerTipousuarioId(int id)  
        {  
            try  
            {  
                var tipousuario = dc.Tbl_TipoUsuario.Single(tipusu => tipusu.TipUsu_Id == id);  
                return tipousuario;  
            }  
            catch (Exception)  
            {  
  
                return null;  
            }  
        }  
  
        private static void Validar(Tbl_TipoUsuario Tbl_TipoUsuario)  
        {
```



```
string mensaje = "";

if (Tbl_TipoUsuario.TipoUsu_Descripcion == "")
{
    mensaje += "INGRESE DESCRIPCION </br>";
}

if (mensaje != "")
{
    throw new ArgumentException(mensaje);
}

}

public static void save(Tbl_TipoUsuario Tbl_TipoUsuario)
{
    Tbl_TipoUsuario.TipoUsu_Id = Secuencial();
    Tbl_TipoUsuario.TipoUsu_Estado = 'A';

    try
    {
        Validar(Tbl_TipoUsuario);
        dc.Tbl_TipoUsuario.InsertOnSubmit(Tbl_TipoUsuario);
        dc.SubmitChanges();
    }
}
```



```
}  
catch (Exception ex)  
{  
  
    throw new ArgumentException("DATOS NO INSERTADOS\n" + ex.Message);  
}  
}  
  
public static void Modificar(Tbl_TipoUsuario Tbl_TipoUsuario)  
{  
    try  
    {  
        dc.SubmitChanges();  
    }  
    catch (Exception ex)  
    {  
  
        throw new ArgumentException("DATOS NO MODIFICADOS\n" + ex.Message);  
    }  
}  
  
public static void eliminar(Tbl_TipoUsuario Tbl_TipoUsuario)  
{  
    try  
    {  
        dc.SubmitChanges();  
    }  
    catch (Exception ex)
```



```
{  
    throw new ArgumentException("No se pudo Eliminar el datos \n" + ex.Message);  
}  
}  
  
public static int Secuencial()  
{  
    int codigo = 0;  
    try  
    {  
        codigo = dc.Tbl_TipoUsuario.Max(tipusu => tipusu.TipUsu_Id + 1);  
        return codigo;  
    }  
    catch (Exception)  
    {  
  
        return 1;  
    }  
}  
}
```

TELEFONOS LOGICA

```
using System;  
using System.Collections.Generic;  
using System.Linq;
```



```
using System.Text;
using System.Data.Linq;
using Acceso_Datos;

namespace Logica_Negocio
{
    public class Telefonos_Logica
    {
        public static CartografiaDataContext dc = new CartografiaDataContext();
        public List <Tbl_Telefonos> ObtenerTelefonos()
        {
            var telefono = dc.Tbl_Telefonos.Select(tel => tel);
            return telefono.ToList();
        }

        public static Tbl_Telefonos ObtenerTelefonoxId(int id)
        {
            try
            {
                var telefono = dc.Tbl_Telefonos.Single(tel => tel.Tel_Id == id);
                return telefono;
            }
            catch (Exception)
            {
                return null;
            }
        }
    }
}
```



```
}

private static void Validar(Tbl_Telefonos Tbl_Telefonos)
{
    string mensaje = "";

    if (Tbl_Telefonos.Tel_Numero == "")
    {
        mensaje += "INGRESE TELEFONO </br>";
    }

    if (mensaje != "")
    {
        throw new ArgumentException(mensaje);
    }
}

public static void save(Tbl_Telefonos Tbl_Telefonos)
{
    Tbl_Telefonos.Tel_Id = Secuencial();
    Tbl_Telefonos.Tel_Estado = 'A';

    try
```




```
{
    Validar(Tbl_Telefonos);
    dc.Tbl_Telefonos.InsertOnSubmit(Tbl_Telefonos);
    dc.SubmitChanges();
}
catch (Exception ex)
{

    throw new ArgumentException("DATOS NO INSERTADOS\n" + ex.Message);
}
}

public static void Modificar(Tbl_Telefonos Tbl_Telefonos)
{
    try
    {
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {

        throw new ArgumentException("DATOS NO MODIFICADOS\n" + ex.Message);
    }
}

public static int Secuencial()
{
    int codigo = 0;
```



```
try
{
    codigo = dc.Tbl_Telefonos.Max(tel => tel.Tel_Id + 1);
    return codigo;
}
catch (Exception)
{
    return 1;
}

}
```

ROL LOGICA

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.Linq;
using Acceso_Datos;

namespace Logica_Negocios
{
```



```
public class Rol_Logica
{
    public static SistemaGeograficoDataContext dc = new
SistemaGeograficoDataContext();
    public static List<Tbl_Rol> ObtenerRol()
    {
        var usuario = dc.Tbl_Rol.Select(rol => rol);
        return usuario.ToList();
    }

    public static Tbl_Rol ObtenerRolxId(int id)
    {
        try
        {
            var Rol = dc.Tbl_Rol.Single(rol => rol.Rol_Id == id);
            return Rol;
        }
        catch (Exception)
        {

            return null;
        }
    }

    private static void Validar(Tbl_Rol Tbl_Rol)
    {
        string mensaje = "";
```



```
if (Tbl_Rol.Rol_Descripcion == "")
{
    mensaje += "INGRESE DESCRIPCION </br>";
}

if (mensaje != "")
{
    throw new ArgumentException(mensaje);
}

}

public static void save(Tbl_Rol Tbl_Rol)
{
    Tbl_Rol.Rol_Id = Secuencial();
    Tbl_Rol.Rol_Estado = 'A';

    try
    {
        Validar(Tbl_Rol);
        dc.Tbl_Rol.InsertOnSubmit(Tbl_Rol);
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {

```



```
        throw new ArgumentException("DATOS NO INSERTADOS\n" + ex.Message);
    }
}

public static void Modificar(Tbl_Rol Tbl_Rol)
{
    try
    {
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("DATOS NO MODIFICADOS\n" + ex.Message);
    }
}

public static void eliminar(Tbl_Rol Tbl_Rol)
{
    try
    {
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("No se pudo Eliminar el datos \n" + ex.Message);
    }
}
```



```
public static int Secuencial()
{
    int codigo = 0;
    try
    {
        codigo = dc.Tbl_Rol.Max(rol => rol.Rol_Id + 1);
        return codigo;
    }
    catch (Exception)
    {
        return 1;
    }
}
}
```

RETENES LOGICA

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.Linq;
using Acceso_Datos;
```



```
namespace Logica_Negocios
```

```
{
```

```
    public class Retenes_Logica
```

```
    {
```

```
        public static SistemaGeograficoDataContext dc = new  
SistemaGeograficoDataContext();
```

```
        public static List<Tbl_Retenes> ObtenerRetenes()
```

```
        {
```

```
            var reten = dc.Tbl_Retenes.Select(ret => ret);
```

```
            return reten.ToList();
```

```
        }
```

```
        public static Tbl_Retenes ObtenerRetenesxId(int id)
```

```
        {
```

```
            try
```

```
            {
```

```
                var reten = dc.Tbl_Retenes.Single(ret => ret.Ret_Id == id);
```

```
                return reten;
```

```
            }
```

```
            catch (Exception)
```

```
            {
```

```
                return null;
```

```
            }
```

```
        }
```

```
        private static void Validar(Tbl_Retenes Tbl_Retenes)
```

```
        {
```



```
string mensaje = "";

if (Tbl_Retenes.Ret_Descripcion == "")
{
    mensaje += "INGRESE DESCRIPCION </br>";
}

if (mensaje != "")
{
    throw new ArgumentException(mensaje);
}

}

public static void save(Tbl_Retenes Tbl_Retenes)
{
    Tbl_Retenes.Ret_Id = Secuencial();
    Tbl_Retenes.Ret_Estado = 'A';

    try
    {
        Validar(Tbl_Retenes);
        dc.Tbl_Retenes.InsertOnSubmit(Tbl_Retenes);
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
    }
```




```
        throw new ArgumentException("DATOS NO INSERTADOS\n" + ex.Message);
    }
}

public static void Modificar(Tbl_Retenes Tbl_Retenes)
{
    try
    {
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("DATOS NO MODIFICADOS\n" + ex.Message);
    }
}

public static void eliminar(Tbl_Retenes Tbl_Retenes)
{
    try
    {
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("No se pudo Eliminar el datos \n" + ex.Message);
    }
}
```



```
public static int Secuencial()
{
    int codigo = 0;
    try
    {
        codigo = dc.Tbl_Retenes.Max(ret => ret.Ret_Id + 1);
        return codigo;
    }
    catch (Exception)
    {
        return 1;
    }
}
}
```

PARAMETROS LOGICA

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.Linq;
using Acceso_Datos;
```



```
namespace Logica_Negocios
{
    public class Parametros_Logica
    {
        public static SistemaGeograficoDataContext dc = new
SistemaGeograficoDataContext();

        public static List<Tbl_Parametros> ObtenerParametros()
        {
            var usuario = dc.Tbl_Parametros.Select(par => par);
            return usuario.ToList();
        }

        public static Tbl_Parametros ObtenerParametrosxId(int id)
        {
            try
            {
                var Parametro = dc.Tbl_Parametros.Single(par => par.Par_Id == id);
                return Parametro;
            }
            catch (Exception)
            {
                return null;
            }
        }

        private static void Validar(Tbl_Parametros Tbl_Parametros)
        {

```



```
string mensaje = "";

if (Tbl_Parametros.Par_Descripcion == "")
{
    mensaje += "INGRESE DESCRIPCION </br>";
}

if (mensaje != "")
{
    throw new ArgumentException(mensaje);
}

}

public static void save(Tbl_Parametros Tbl_Parametros)
{
    Tbl_Parametros.Par_Id = Secuencial();
    Tbl_Parametros.Par_Estado = 'A';

    try
    {
        Validar(Tbl_Parametros);
        dc.Tbl_Parametros.InsertOnSubmit(Tbl_Parametros);
        dc.SubmitChanges();
    }
    catch (Exception ex)
```



```
{

    throw new ArgumentException("DATOS NO INSERTADOS\n" + ex.Message);

}

}

public static void Modificar(Tbl_Parametros Tbl_Parametros)
{
    try
    {
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {

        throw new ArgumentException("DATOS NO MODIFICADOS\n" + ex.Message);

    }
}

public static void eliminar(Tbl_Parametros Tbl_Parametros)
{
    try
    {
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {

        throw new ArgumentException("No se pudo Eliminar el datos \n" + ex.Message);
```



```
}  
}  
  
public static int Secuencial()  
{  
    int codigo = 0;  
  
    try  
    {  
        codigo = dc.Tbl_Parametros.Max(par => par.Par_Id + 1);  
        return codigo;  
  
    }  
    catch (Exception)  
    {  
  
        return 1;  
    }  
  
}  
}
```

DEPARTAMENTO LOGICA

```
using System;
```



```
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.Linq;
using Acceso_Datos;

namespace Logica_Negocios
{
    public class Departamento_Logica
    {
        public static SistemaGeograficoDataContext dc = new
SistemaGeograficoDataContext();
        public static List<Tbl_Departamento> ObtenerDepartamentos()
        {
            var usuario = dc.Tbl_Departamento.Select(dep => dep);
            return usuario.ToList();
        }

        public static Tbl_Departamento ObtenerDepartamentoxId(int id)
        {
            try
            {
                var departamento = dc.Tbl_Departamento.Single(dep => dep.Dep_Id == id);
                return departamento;
            }
            catch (Exception)
            {

```



```
        return null;
    }
}

private static void Validar(Tbl_Departamento Tbl_Departamento)
{
    string mensaje = "";

    if (Tbl_Departamento.Dep_Descripcion == "")
    {
        mensaje += "INGRESE DESCRIPCION </br>";
    }

    if (mensaje != "")
    {
        throw new ArgumentException(mensaje);
    }
}

public static void save(Tbl_Departamento Tbl_Departamento)
{
    Tbl_Departamento.Dep_Id = Secuencial();
    Tbl_Departamento.Dep_Estado = 'A';

    try
```




```
{
    Validar(Tbl_Departamento);
    dc.Tbl_Departamento.InsertOnSubmit(Tbl_Departamento);
    dc.SubmitChanges();
}
catch (Exception ex)
{
    throw new ArgumentException("DATOS NO INSERTADOS\n" + ex.Message);
}
}

public static void Modificar(Tbl_Departamento Tbl_Departamento)
{
    try
    {
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("DATOS NO MODIFICADOS\n" + ex.Message);
    }
}

public static void eliminar(Tbl_Departamento Tbl_Departamento)
{
    try
```



```
{
    dc.SubmitChanges();
}
catch (Exception ex)
{
    throw new ArgumentException("No se pudo Eliminar el datos \n" + ex.Message);
}
}

public static int Secuencial()
{
    int codigo = 0;
    try
    {
        codigo = dc.Tbl_Departamento.Max(dep => dep.Dep_Id + 1);
        return codigo;
    }
    catch (Exception)
    {
        return 1;
    }
}
}
```

COORDENADAS LÓGICA



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.Linq;
using Acceso_Datos;

namespace Logica_Negocios
{
    public class Coordenadas_Logica
    {
        public static SistemaGeograficoDataContext dc = new
SistemaGeograficoDataContext();
        public static List<Tbl_Coordenadas> ObtenerCoordenadas()
        {
            var coordenada = dc.Tbl_Coordenadas.Select(cor => cor);
            return coordenada.ToList();
        }

        public static Tbl_Coordenadas ObtenerCoordenadasxId(int id)
        {
            try
            {
                var coordenada = dc.Tbl_Coordenadas.Single(cor => cor.Coor_Id == id);
                return coordenada;
            }
            catch (Exception)
```



```
{  
  
    return null;  
}  
}  
  
private static void Validar(Tbl_Coordenadas Tbl_Coordenadas)  
{  
    string mensaje = "";  
  
    if (Tbl_Coordenadas.Coor_Latitud == "")  
    {  
        mensaje += "INGRESE LATITUD </br>";  
    }  
  
    if (mensaje != "")  
    {  
        throw new ArgumentException(mensaje);  
    }  
}  
  
public static void save(Tbl_Coordenadas Tbl_Coordenadas)  
{  
    Tbl_Coordenadas.Coor_Id = Secuencial();  
    Tbl_Coordenadas.Coor_Estado = 'A';
```



```
try
{
    Validar(Tbl_Coordenadas);
    dc.Tbl_Coordenadas.InsertOnSubmit(Tbl_Coordenadas);
    dc.SubmitChanges();
}
catch (Exception ex)
{
    throw new ArgumentException("DATOS NO INSERTADOS\n" + ex.Message);
}

public static void Modificar(Tbl_Coordenadas Tbl_Coordenadas)
{
    try
    {
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("DATOS NO MODIFICADOS\n" + ex.Message);
    }
}
```



```
public static void eliminar(Tbl_Coordenadas Tbl_Coordenadas)
{
    try
    {
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("No se pudo Eliminar el datos \n" + ex.Message);
    }
}

public static int Secuencial()
{
    int codigo = 0;
    try
    {
        codigo = dc.Tbl_Coordenadas.Max(cor => cor.Coor_Id + 1);
        return codigo;
    }
    catch (Exception)
    {
        return 1;
    }
}
```



```
}  
}
```

CARGO LOGICA

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Data.Linq;  
using Acceso_Datos;  
  
namespace Logica_Negocios  
{  
    public class Cargo_Logica  
    {  
        public static SistemaGeograficoDataContext dc = new  
SistemaGeograficoDataContext();  
        public static List<Tbl_Cargo> ObtenerCargo()  
        {  
            var usuario = dc.Tbl_Cargo.Select(carg => carg);  
            return usuario.ToList();  
        }  
  
        public static Tbl_Cargo ObtenerCargoxId(int id)  
        {  
            try
```



```
{  
    var cargo = dc.Tbl_Cargo.Single(carg => carg.Car_Id == id);  
    return cargo;  
}  
catch (Exception)  
{  
  
    return null;  
}  
}  
  
private static void Validar(Tbl_Cargo Tbl_Cargo)  
{  
    string mensaje = "";  
  
    if (Tbl_Cargo.Car_Descripcion == "")  
    {  
        mensaje += "INGRESE DESCRIPCION </br>";  
    }  
  
    if (mensaje != "")  
    {  
        throw new ArgumentException(mensaje);  
    }  
}
```




```
public static void save(Tbl_Cargo Tbl_Cargo)
{
    Tbl_Cargo.Car_Id = Secuencial();
    Tbl_Cargo.Car_Estado = 'A';

    try
    {
        Validar(Tbl_Cargo);
        dc.Tbl_Cargo.InsertOnSubmit(Tbl_Cargo);
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {

        throw new ArgumentException("DATOS NO INSERTADOS\n" + ex.Message);
    }
}

public static void Modificar(Tbl_Cargo Tbl_Cargo)
{
    try
    {
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {

```



```
        throw new ArgumentException("DATOS NO MODIFICADOS\n" + ex.Message);
    }
}

public static void eliminar(Tbl_Cargo Tbl_Cargo)
{
    try
    {
        dc.SubmitChanges();
    }
    catch (Exception ex)
    {
        throw new ArgumentException("No se pudo Eliminar el datos \n" + ex.Message);
    }
}

public static int Secuencial()
{
    int codigo = 0;
    try
    {
        codigo = dc.Tbl_Cargo.Max(carg => carg.Car_Id + 1);
        return codigo;
    }
    catch (Exception)
    {

        return 1;
    }
}
```



}

}

}

}

Recurso Económico

DETALLE	COSTO
COMPUTADORAS	3.000
PAPEL BOND	100
IMPRESORA	400
SISTEMA DE TINTA	200
VARIOS	300
TOTAL	4.000

Recurso Humano

Nombre	Descripción
XAVIER OCHOA	ALUMNO
ING. PAUL CURICHO	TUTOR