

Universidad de Sancti Spíritus “José Martí Pérez”

Facultad de Ingeniería

Carrera Ingeniería Informática



Título: “Sistema de Gestión para la Empresa de Transporte de la Construcción de Sancti Spíritus.”

Trabajo de diploma para optar por el título de Ingeniería en
Informática.

Autora: Yangzet de la C. Valdés Paneca

Tutores: MSc. Niriam Peña Valdés

Ing. Yobisley Bernal Turiño

Consultantes: Lic. Blanca Hernández Concepción

MSc. Sandor Gómez Pérez

Curso: 2012-2013

Pensamiento

Pensamiento

“El estudio engrandece al hombre y el trabajo lo dignifica.”

Anónimo.

Dedicatoria

*Dedico este trabajo a mis padres, mi hermano, mi
abuela, mi bella sobrina y a mi cosito.*

Agradecimientos

- *Agradezco en primer lugar a Dios pues su presencia en mi vida me ha llevado a ser quien soy hoy.*
- *A mis padres por haber dado día a día todo su amor y entrega esta niña que poco a poco se ha convertido en una mujer lista para enfrentar la vida.*
- *A mi papá bello y hermoso que hoy no está presente de cuerpo pero si en espíritu y gracias a él he puesto todas mis fuerzas en esta carrera y he podido avanzar con pasos agigantados desde el día que decidí estudiar informática en la primaria ya que ese ímpetu, esa dedicación, inteligencia, valentía, honestidad, humildad, y no terminaría nunca de describirlo que lo caracterizaban fue lo que me ayudó a comprender que cuando le pones esfuerzo a todas las cosas que hagas las lograrás y así te sentirás satisfecho contigo y con los demás. Para ti papito mío, te amo.*
- *A mi madre linda y hermosa por todos sus desvelos, sus cariños, sus preocupaciones, sus consejos que me han ayudado a llevar de una forma más práctica esta carrera pues siempre estuvo ahí brindándome su pecho y sus brazos para acogerme cuando algo no salía como quería.*
- *A mi hermano que siempre ha sido mi ídolo, mi guía, mi amigo, mi ejemplo a seguir pues me ha enseñado a amar a Dios sobre todas las cosas, a ver las cosas como son, a llamar a todo por su nombre, a luchar por lo que quiero, a ser una mejor persona, y simplemente a ser hermana sobre toda situación. Titi te amo mucho.*
- *A mi abuelita linda, mi tribilingua por ser tan cariñosa conmigo y ayudarme siempre cuando se lo he pedido.*
- *A la sobrina linda y hermosa que a pesar que llegó hace poco ha llenado de alegría y felicidad la vida de todos en la casa.*
- *A mi cosito del alma, mi amor verdadero, mi novio, mi futuro esposo, mi amigo, por amarme desde el primer día que comenzó nuestra relación, por haberme dado la fuerza para enfrentar todas las adversidades que fueron apareciendo en el camino, por no dejarme caer nunca, por cada consejo que me dio en el momento preciso, por aceptarme como soy, por ser la persona más hermosa del mundo.*
- *A mi cuñada Yanet por su ayuda incondicional en cada momento.*

- *A mis suegros por haberme acogido como una hija más y ayudarme siempre que lo he necesitado.*
- *A mi cuñada Yeni por su apoyo en todo momento.*
- *A mis amigas Geisa, Maibel, Lilliana y Gretel por darme ánimo y apoyo en cada uno de los momentos que lo necesité.*
- *A Ivón y Reinaldo por ser como padres y ejemplos para mí.*
- *A mi abuela y madre postizas Consuelo y Miriam por su cariño y amor que me han mostrado siempre. Al igual que Gustavo, Ketty, Mario y Mario Alfredo por su apoyo.*
- *A las Siervas de San José por cada consejo y amor entregado desde que las conocí.*
- *Al grupo de espiritualidad josefina por sus rezos y apoyo en todo momento.*
- *A las hermanas del Sagrado Corazón de Jesús por su apoyo y comprensión en todo momento.*
- *A los jóvenes de la Pastoral Juvenil por su ánimo y comprensión especialmente Ángel, Lisbet, Jorge, Ernesto y Marcela.*
- *A mis profesores por la formación que me dieron durante estos 5 años.*
- *A mis compañeros de aula por esto 5 años, especialmente a Xiomara por ser mi amiga y hermana en todo este tiempo.*
- *A mis tutores Niriam y Yobisley por su amistad y sus consejos para lograr hacer este trabajo. Al igual que mis consultantes por su ayuda en todo momento.*
- *A todas mis amistades por su ánimo y apoyo.*
- *A todos aquellos que de una forma u otra han contribuido con su apoyo.*

A todos, muchas gracias.

Yangzet Valdés

Resumen

Este trabajo se desarrolla en la Empresa de Transporte de la Construcción de Sancti Spíritus, donde actualmente el proceso de gestión de la información relacionada con la transportación de materiales de la construcción se viene desarrollando a través del sistema de escritorio de gestión de bases de datos Microsoft Access siendo un gestor que se utiliza para pequeñas organizaciones y sus servicios no alcanzan para los objetivos que persigue la empresa. Es por ello que el objetivo de esta investigación es desarrollar un software que facilite la gestión de la información en esta entidad. Las etapas llevadas a cabo en la elaboración del software son las planteadas por el Proceso Unificado de Desarrollo (RUP), utilizándose como lenguaje el “Lenguaje Unificado de Modelado” (UML) y como herramienta para el análisis y diseño: el Rational Rose Enterprise Edition. Para implementar el sistema se utilizó Java como lenguaje de programación lo que proporcionó un código eficiente, para la persistencia de los datos se utilizó PostgreSQL versión 8.4 y como herramienta de programación el Netbeans versión 6.8

Abstract

This work is developed in the the Construction Transport enterprise of Sancti Spíritus, where at the momento, the process of information management related to the transportation of materials of the construction is developing through the desk system of administration of databases Microsoft Access being an agent that is used for small organizations and its services don't reach for the objectives that the company pursues. For this reason the objective of this investigation is to develop a software that facilitates the information management in this entity. The stages carried out in the elaboration of the software are those outlined by the Unified Process of Development (RUP), using the Unified Language of Modeling" (UML) as language and as tool for the analysis and design: the Rational Rose Enterprise Edition. To implement the system Java was used like programming language which provided an efficient code, for the persistence of the data PostgreSQL version 8.4 was used and as programming tool the Netbeans version 6.8.

Índice general

Introducción	1
1.1 Introducción.....	5
1.2 La gestión de información	5
1.3 Empresas de Transporte de la Construcción.....	7
1.4 Fundamentación de la metodología utilizada.....	7
1.4.1 Proceso Unificado Racional.....	7
1.4.2 Lenguaje Unificado de Modelado (UML).....	8
1.4.3 Rational Rose	8
1.5 Tendencias y tecnologías actuales.....	8
1.5.1 Internet	9
1.5.2 Migración hacia software libre.....	9
1.5.3 Software Libre.....	10
1.5.4 Arquitectura de tres capas	11
1.6 Lenguajes de programación	14
1.7 Herramientas de desarrollo	15
1.8 Sistemas Gestores de Bases de Datos	18
Conclusiones.....	21
2.1 Introducción.....	23
2.2 Modelo del negocio.	23
2.1.1 Identificación de los procesos del negocio	24
2.1.2 Reglas del negocio	24
2.1.3 Modelo de casos de uso del negocio	25
2.1.4 Actores del negocio	25
2.1.5 Diagrama de casos de uso del negocio	25

2.1.6 Trabajadores del negocio.....	26
2.1.7 Descripción de los casos de uso del negocio.....	27
2.1.8 Diagramas de actividades.....	29
2.1.9 Modelo de objetos del negocio.....	32
2.2 Requerimientos.....	33
2.2.1 Requerimientos funcionales.....	33
2.2.2 Requerimientos no funcionales.....	35
2.3 Modelo del sistema.....	37
2.3.1 Modelo de casos de uso del sistema.....	37
2.3.2 Actores del sistema.....	37
2.3.3 Casos de uso del sistema.....	38
2.3.4 Diagramas de casos de uso del sistema.....	39
Conclusiones.....	43
Capítulo III: Implementación de un software que contribuya a la gestión de información de la Empresa de Transporte de Sancti Spíritus.....	44
Introducción.....	44
3.1 Diagrama de clases del diseño.....	44
3.2 Diseño de la base de datos.....	56
3.2.1 Diagrama de clases persistentes.....	56
3.2.2 Modelo de datos.....	57
3.3 Principios de diseño.....	58
3.3.1 Diseño de la interfaz del sistema.....	58
3.3.2 Tratamiento de errores.....	59
3.3.3 Concepción general de la ayuda.....	59
3.3.4 Seguridad.....	59
3.4 Estándares de codificación.....	59

3.5 Modelo de implementación.....	60
3.5.1 Diagrama de despliegue	60
3.5.2 Diagrama de componentes	61
Conclusiones.....	62
Conclusiones Generales.....	63
Recomendaciones	64
Anexos	68

Índice de figuras

Figura 1 Esquema Modelo – Vista – Controlador.....	13
Figura 2 Diagrama de casos de uso del negocio	26
Figura 3 Diagrama de actividades: Realizar contrato de servicio	30
Figura 4 Diagrama de actividades: Llenar información de la Carta de Porte	31
Figura 5 Diagrama de actividades: Confeccionar informes.	32
Figura 6 Diagrama de casos de uso por paquetes.....	39
Figura 7 Diagrama de casos de uso: Paquete Seguridad	40
Figura 8 Diagrama de casos de uso: Paquete Administración	41
Figura 9 Diagrama de casos de uso: Paquete Catálogo	42
Figura 10 Diagrama de casos de uso: Paquete Reportes	43
Figura 11 Diagrama de clases del diseño Autenticar usuario.....	45
Figura 12 Diagrama de clases del diseño Cambiar contraseña	46
Figura 13 Diagrama de clases del diseño Parte diario	47
Figura 14 Diagrama de clases del diseño Gestionar chofer	48
Figura 15 Diagrama de clases del diseño Gestionar destino	49
Figura 16 Diagrama de clases del diseño Gestionar empresa	50
Figura 17 Diagrama de clases del diseño Gestionar equipo	51
Figura 18 Diagrama de clases del diseño Gestionar municipio	52
Figura 19 Diagrama de clases del diseño Gestionar origen	53
Figura 20 Diagrama de clases del diseño Gestionar producto	54
Figura 21 Diagrama de clases del diseño Gestionar rol	55
Figura 22 Diagrama de clases del diseño Gestionar usuario	56
Figura 23 Diagrama de clases persistentes	57

Figura 24 Modelo de Datos.....	58
Figura 25 Diagrama de despliegue	60
Figura 26 Diagrama de componentes.....	61

Índice de Tablas

Tabla 1 Actores del negocio	25
Tabla 2 Trabajadores del negocio	27
Tabla 3 Descripción del caso de uso: Realizar contrato de servicio.	27
Tabla 4 Descripción del caso de uso: Confeccionar informes.....	28
Tabla 5 Descripción del caso de uso: Llenar información de la Carta de Porte	29
Tabla 6 Actores del sistema.	37
Tabla 6 Descripción del diagrama de componentes	62

Introducción

El hecho de desplazarse de un lugar a otro es consustancial al hombre. Desde tiempos primitivos, éste ha buscado medios para moverse con más comodidad y con el progreso del comercio, sistemas para desplazar sus mercancías. El medio de transporte más antiguo fue el fluvial, las rudimentarias canoas. En tierra, disímiles frutos del ingenio humano lo ayudaron por muchos años hasta la llegada de los carros de tiro. La Revolución Industrial permitió su redimensionamiento he indetenible progreso. Hoy tenemos rápidos ferrocarriles y automóviles, así como enormes embarcaciones y medios aéreos transoceánicos que ayudan al movimiento de personas y al entorno comercial.

Hoy en día el sector del transporte es esencial para el funcionamiento de los países, siendo un servicio de interés público que incluye todos los medios e infraestructura implicados en este desplazamiento, así como los servicios de recepción, entrega y manipulación de cada una de las mercancías. Infraestructuras viarias: carreteras, vías de ferrocarril, puertos y aeropuertos, son potenciados por muchos gobiernos y puestos a disposición de todos. Para cualquier zona subdesarrollada la construcción de una carretera es una garantía, y una condición, de desarrollo.

La Empresa de Transporte de la Construcción (ETC) surge como consecuencia de la creación del Grupo Empresarial de la Construcción de Sancti Spíritus (GECSS). En sus inicios se encargaba del transporte y comercialización de los materiales de la construcción para las diferentes empresas constructoras del GECSS; más tarde como política del Ministerio de la Construcción (MICONS) dejó de formar parte de su objeto social la comercialización y desde entonces sólo se dedica a la transportación de materiales de la construcción, siendo su objeto social prestar servicios de transporte especializado con una alta calidad y reconocimiento del cliente.

En estos momentos dicha empresa forma parte del reordenamiento del transporte y está comprometida con varios programas de interés de la provincia. El punto de inicio en el procedimiento de trabajo de esta entidad es la demanda de materiales de la construcción que emite cada obra y a partir del cual la empresa conforma su plan de transportación del mes. Cada viaje ha de estar avalado por un modelo denominado Carta de Porte que acompañará al chofer durante todo el recorrido desde su salida de la Empresa, llegada, espera y carga en el centro de carga; regreso llegada, espera y descarga en el centro de descarga y vuelta a la

empresa donde entrega dicha carta de Porte en la que se ha ido recogiendo información en los diferentes puntos del recorrido.

A partir de la Carta de Porte la dirección de la empresa conoce e informa algunos parámetros como producción en físico, en valores, consumo de combustible, entre otros de interés a diferentes instancias que lo solicitan (GECSS, Partido, Gobierno, etc.). La informatización los procesos permitió en principio la gestión de esta información mediante tablas Microsoft Office EXCEL pero resultaba prácticamente imposible mantener a todos informados con la inmediatez y confiabilidad requeridas. Es por ello que surge la idea de utilizar un sistema de gestión de bases de datos en este caso Microsoft Office ACCESS dando lugar a la creación del Sistema de Gestión del Transporte. Por medio de entrevistas realizadas a especialistas y directivos que interactúan con esta herramienta se pudo constatar que no les ha facilitado el proceso de gestión de la información, de ella se han derivado una serie de problemas, entre los más significativos están:

- Dificil acceso a los datos
- Redundancia e inconsistencia de los datos.
- Aislamiento de datos.
- Problemas de integridad.
- Problemas de Seguridad.

Lo expuesto anteriormente, nos permite identificar como **problema de investigación:** Dificultades en el procesamiento de la información en la Empresa Transporte de la Construcción de Sancti Spíritus.

Cuya causa fundamental se atribuye a la utilización de un sistema que utiliza software propietario (Microsoft Office ACCESS), limitando la readecuación del mismo a los intereses y necesidades actuales de la empresa.

El **objetivo general** es desarrollar un sistema informático basado en software libre que contribuya a la gestión de la información de la Empresa de Transporte de la Construcción de Sancti Spíritus.

Para dar cumplimiento al objetivo general surgen las siguientes **preguntas de investigación:**

- 1- ¿Cuáles son los fundamentos teóricos y metodológicos que sustentan la creación de un software para la gestión de la información en las Empresas de Transporte de la Construcción?
- 2- ¿Cómo diseñar una aplicación que contribuya a la gestión de la información de la Empresa de Transporte de Sancti Spíritus?
- 3- ¿Cómo implementar una aplicación que contribuya a la gestión de la información de la Empresa de Transporte de Sancti Spíritus?

Para dar respuesta a las preguntas científicas surgieron las siguientes **tareas de investigación**:

- 1- Determinar los fundamentos teóricos y metodológicos que sustentan la creación de un software para la gestión de la información en las Empresas de Transporte de la Construcción.
- 2- Diseñar una aplicación que contribuya a la gestión de la información de la Empresa de Transporte de Sancti Spíritus.
- 3- Implementar una aplicación que contribuya a la gestión de la información de la Empresa de Transporte de Sancti Spíritus.

Estructura del documento

Introducción: expresa los aspectos más generales de la investigación del diseño teórico y metodológico.

Capítulo I: Fundamentación teórica y metodológica que sustentan la realización de un software que va dirigido a la gestión de información de una empresa.: se explican los conceptos asociados al dominio del problema, se describen brevemente las tendencias y tecnologías actuales a tener en cuenta para implementar la aplicación, además de las metodologías y herramientas existentes justificando la elección de cada una, para el análisis, diseño, e implementación de la aplicación.

Capítulo II: Diseño de una herramienta basada en software libre que contribuya a la gestión de información de la Empresa de Transporte de Sancti Spíritus: este capítulo describe todos los elementos que componen el modelo del negocio. Se identifican los actores y

trabajadores que intervienen, el diagrama de casos de uso y el de modelo de objetos, así como se determinan los requerimientos funcionales y no funcionales del sistema. Además se definen los casos de uso del sistema y los diagramas correspondientes a cada caso de uso del sistema a automatizar.

Capítulo III: Implementación del software propuesto para la gestión de la información de la Empresa de Transporte de Sancti Spíritus: en el capítulo se realiza la descripción de cada caso de uso del sistema, se describe el diagrama de clases de diseño, el diagrama de clases persistentes, el modelo de datos y la descripción de cada una de las tablas que contiene la base de datos; se enuncian también los principios de diseño, determinando los estándares usados en las diferentes interfaces de la aplicación, la concepción general de la ayuda, el tratamiento de excepciones y cómo es manejada la seguridad. Además, se describe la implementación incluyendo el diagrama de despliegue y el diagrama de componentes.

Capítulo I: Fundamentación teórica y metodológica que sustentan la realización de un software que va dirigido a la gestión de información de una empresa.

1.1 Introducción

El presente capítulo incluye la base de la fundamentación teórica del tema a desarrollar. Describe los principales conceptos asociados al problema, la estructura organizativa de la Empresa de Transporte de la Construcción de Sancti-Spíritus. Se analiza además el objeto de estudio y la situación problemática en la que se encuentra inmersa. Se realiza un estudio de las distintas metodologías, tecnologías, herramientas actuales y se determinan las que serán utilizadas en el desarrollo del sistema.

1.2 La gestión de información

Por la importancia que representa la recuperación de información, se considera imprescindible profundizar en los elementos que la caracterizan. En literaturas revisadas se encontraron múltiples definiciones de gestión.

Según Ponjuán Dante, (1998), gestión: es el proceso mediante el cual se obtiene, despliega o utiliza una variedad de recursos básicos para apoyar los objetivos de la organización. Por otra parte la Real Academia Española (2001) lo define como: acción y efecto de gestionar o de administrar.

Por otra parte Salinero plantea que “es el proceso emprendido por una o más personas para coordinar las actividades laborales de otros individuos” (Bartle, 2009).

Según Marecos “gestionar es coordinar todos los recursos disponibles para conseguir determinados objetivos, implica amplias y fuertes interacciones fundamentalmente entre entorno, las estructuras, el proceso y los productos que se desean obtener” (Bartle, 2009).

La información es comunicación o adquisición de conocimientos que permiten ampliar o precisar los que se poseen sobre una materia determinada.

Es importante señalar que usar la información implica disponer de ella en el lugar y el momento preciso, contar con una clasificación de la información, que por determinadas características, se sepa puede constituir recurso dentro de la institución, a partir de la correspondencia que su uso aporte al cumplimiento de la misión y los objetivos que se haya trazado la organización no solo ahorra tiempo y esfuerzo, como siempre se ha planteado, sino que proporciona una correcta toma de decisiones que es, en definitiva la clave de todo éxito. El desarrollo actual de los sistemas informáticos y la telemática, facilitan y agilizan el acceso a la información electrónica, por lo que se pueden ofrecer servicios con valor agregado, pero al utilizar esta tecnología, genera la necesidad de diseñar un software para proporcionar la organización, acceso, uso y difusión con mayor rapidez.

La gestión de la información, comprende las actividades relacionadas con la obtención de información adecuada, a un precio adecuado, en tiempo y lugar adecuado, para tomar la decisión adecuada " (Ajá, 2006), Capote Marrero la define como el proceso que se encarga de suministrar los recursos necesarios para la toma de decisiones, así como para mejorar los procesos, productos y servicios de la organización" (Capote, 2006).

"La gestión de la información tiene siempre como objetivo último optimizar recursos a través de un adecuado análisis de las necesidades de información para que la organización y/o sus usuarios puedan alcanzar sus metas" (Chaín, 2007).

Otro autor considera que la gestión de la información es un proceso que incluye operaciones de extracción, manipulación, tratamiento, depuración, conservación, acceso y/o colaboración de la información adquirida por una organización a través de diferentes fuentes y que gestiona el acceso y los derechos de los usuarios sobre el sistema (Curto, 2006).

Corresponde a los profesionales de la información la enseñanza de las destrezas en el acceso y uso de información más importante. Por lo que la autora se acoge a lo planteado por Celia Chaín Navarro sobre la necesidad de organizar y optimizar la información para tenerla a disposición del usuario. No es solo la cantidad sino la calidad de la misma lo que va a determinar la calidad de los resultados del trabajo de investigación o el trabajo en general.

Aplicando los anteriores conceptos de gestión de la información a la información concerniente al proceso productivo en la empresa de transporte de la construcción de Sancti Spiritus se podrá lograr el nivel de especificación y organización requerido en este caso, además con toda la información contenida en un mismo volumen el acceso a los datos tendrá mayor calidad y claridad y se ganará en reducción del tiempo de procesamiento de la información así como en errores introducidos por el usuario.

1.3 Empresas de Transporte de la Construcción

La Empresa de Transporte de la Construcción realiza diversas funciones entre ellas se encuentran:

- Prestación de servicios para la transportación de materiales de la construcción donde cada viaje ha de estar avalado por un modelo denominado Carta de Porte que acompañará al chofer durante todo el recorrido desde su salida de la Empresa, llegada, espera y carga en el centro de carga; regreso llegada, espera y descarga en el centro de descarga y vuelta a la empresa donde entrega dicha Carta de Porte en la que se ha ido recogiendo información en los diferentes puntos del recorrido. A partir de esta última la dirección de la empresa conoce e informa algunos parámetros de interés como producción en físico, en valores, consumo de combustible, y varios más a diferentes instancias que lo solicitan (GECSS, Partido, Gobierno, etc.).
- Cumplimiento de las demandas de cada una de las obras constructoras.
- Cumplimiento del plan establecido en el mes.
- Cuidado y conservación del medio ambiente.(Cubagob, 2013).

1.4 Fundamentación de la metodología utilizada

1.4.1 Proceso Unificado Racional

Para el diseño metodológico de la presente investigación, se decidió emplear la metodología “Proceso Unificado Racional”, RUP (Rational Unified Process), que es la más utilizada para el análisis, diseño, implementación y documentación de sistemas orientados a objetos. Está dirigida por casos de usos, centrada en la arquitectura, y es iterativa e incremental. Es una metodología con la capacidad de adaptación a cualquier proyecto, cuyo objetivo es producir software de alta calidad, cumplimentando los requerimientos de los usuarios dentro de una planificación y presupuesto establecidos (Rumbaugh, Booch, & Jacobson, 2006).

El RUP es un conjunto de metodologías adaptables al contexto y necesidades de cada organización. Describe como aplicar enfoques para el desarrollo del software, llevando a cabo unos pasos para su realización. Se centra en la producción y mantenimiento de modelos del sistema (Rumbaugh, Booch & Jacobson, 2000).

1.4.2 Lenguaje Unificado de Modelado (UML)

Es una herramienta de lenguaje visual para modelación de sistemas (comprendiendo hardware y software) que permite generar diagramas organizados que capten todas las ideas de un proceso, siendo su principal finalidad que sea fácil de comprender para todas las personas involucradas en el desarrollo del sistema. Esto tiene la ventaja que al realizar un plan de diseño organizado a través de un cuidadoso análisis de las necesidades del cliente, se puede contar con un diseño sólido, el cual, puede ser modificado sin mayores problemas. Por consiguiente el objetivo del Proceso Unificado es guiar a los desarrolladores en la implementación y distribución eficiente de sistemas que se ajusten a las necesidades de los clientes. UML es ante todo un lenguaje. Y como todo lenguaje proporciona vocabulario y reglas para permitir la comunicación. En este caso, el lenguaje se centra en la representación gráfica de un sistema. (Rumbaugh, Booch & Jacobson, 2000)

1.4.3 Rational Rose

Para facilitar la documentación de la presente investigación, se emplea la herramienta de ingeniería de software asistida por computadora, CASE (Computer Assisted Software Engineering), Rational Rose Enterprise. Potente aplicación que integra todos los elementos de la metodología RUP y la notación UML, para cubrir el análisis, diseño, implementación y prueba del ciclo de vida básico de un sistema informático. Permitiendo realizar la generación de diagramas y códigos, la automatización de sistemas, y la ingeniería inversa.

1.5 Tendencias y tecnologías actuales

El estudio de las tecnologías actuales se ha convertido en uno de los factores claves en el desarrollo de cualquier sistema informático por muy sencillo que este resulte. Representa uno de los aspectos determinantes en el funcionamiento de las empresas e instituciones. En la medida en que el concepto calidad tecnológica sea mayor, se logrará alcanzar los objetivos trazados de una forma eficiente y efectiva. Se necesita realizar una correcta selección de las tecnologías a emplear, en dependencia de las necesidades y recursos propios con los que se cuenta. Una correcta selección permitirá lograr un producto final, con calidad y eficiencia (Buch, 1999).

En los epígrafes que vienen a continuación se pretende ofrecer un camino informativo a través de las tecnologías que se encuentran entorno de las requeridas para el desarrollo del sistema propuesto, analizando las ventajas y desventajas que las caracterizan, con el fin de poder justificar el por qué del lenguaje, y otros software utilizados. Además de la fundamentación del por qué de la propuesta que estamos haciendo para el desarrollo de esta investigación

1.5.1 Internet

Internet es una red de computadoras a nivel mundial, con vasto repertorio de información y recursos a los que se puede acceder desde una computadora (Chaín, 2007).

En casi todas las bibliografías se describe Internet como la "red de redes" o como una "colección de redes enlazadas". Ambas descripciones son adecuadas, pero sólo parcialmente (Chaín, 2007).

Según Chaín (2007) internet representa, además, la comunidad de intereses sobre un tema y la herramienta más útil de la actualidad para intercambiar información de una manera rápida y efectiva. En la red de redes se puede encontrar información sobre cualquier tema de interés; resulta una Biblioteca Universal, en la que el usuario participa directamente en el proceso de selección, adquisición y procesamiento de la información de su interés, a través de los recursos y programas existentes.

1.5.2 Migración hacia software libre

Migración: como término general se le denomina a todo desplazamiento de la población (humana o animal) que se produce desde un lugar de origen a otro destino y lleva consigo un cambio de la residencia habitual en el caso de las personas o del hábitat en el caso de las especies animales migratorias (Dehesa,2008).

Pero en el mundo de la informática se utiliza para el proceso consistente en hacer que los datos y las aplicaciones existentes funcionen en una computadora, software o sistema operativo diferentes, tomando este mayor auge con la aparición del software libre que vino ayudar a muchos los países, principalmente los países subdesarrollados puesto que en la mayoría de estos el pago de las licencias de los software privativos consume una parte importante de sus presupuestos económicos.

1.5.3 Software Libre

Hasta la década de los años 70 los software eran productos añadidos que los vendedores de computadoras de la época ponían a disposición de los usuarios, o sea, aun no habían tomado carácter de mercancía. Ello propiciaba que los programadores compartieran libremente sus programas unos con otros.

A finales de la década de 1970, comenzaron la imposición de restricciones a los usuarios mediante el uso de acuerdos de licencia, lo cual llegó a su máxima expresión con la utilización a partir de los 80 de los sistemas operativos privativos y sus impedimentos a los usuarios para la realización de transformaciones a los programas que tomarían el nombre de software propietarios.

El usuario, a pesar de dominar los conocimientos necesarios para modificar el software y no poseer interés de lucro, debía restringirse a los contratos, que únicamente le permitían informar a la empresa desarrolladora los problemas encontrados para que esta los solucionara.

A raíz de las dificultades que presenta el software propietario para el desempeño del trabajo de las empresas, en 1984 Richard Stallman comenzó sus trabajos para la creación del software libre. Creó un sistema operativo compatible con Unix para dotarle así de portabilidad y facilitar el cambio a los usuarios de Unix. El nombre de GNU fue elegido según una tradición de los hackers, como un acrónimo recursivo de «GNU's Not Unix»(GNU no es Unix).

En 1985 fundó la Free Software Foundation (FSF) para buscar fuentes de financiamiento y continuar la obra empezada: la construcción de software libre, que contribuye a eliminar las restricciones sobre la copia, redistribución, entendimiento y modificación de programas de computadoras y da cierta libertad de ejecutar el programa sea cual sea el propósito, modificar el programa para ajustarlo a nuestras necesidades y para que se trate de una libertad efectiva en la práctica, debemos tener acceso al código fuente, dado que sin él la tarea de incorporar cambios en un programa es extremadamente difícil, la libertad de redistribuir copias, ya sea de forma gratuita o a cambio del pago de un precio y de distribuir versiones modificadas del programa, de tal forma que la comunidad pueda aprovechar las mejoras introducidas (Stallman, 2004).

Proporcionando así disímiles ventajas entre ellas se encuentran su bajo costo de adquisición, pues evita la interminable compra de licencias; la innovación tecnológica, pues permite la

modificación de los programas por parte de cada usuario según sus conocimientos y con ello se decide la dirección del desarrollo del software.

Otro aspecto positivo es la adaptación del software, lo cual posibilita a las empresas e industrias la materialización de un programa personalizado para la realización de un trabajo específico.

Este presenta cuatro libertades (Free Software Fundation, 2008):

- Libertad para ejecutar el programa en cualquier Sitio, con cualquier propósito y para siempre.
- Libertad para estudiarlo y adaptarlo a nuestras necesidades. Esto exige el acceso al código fuente.
- Libertad de redistribución, de modo que se nos permita colaborar con vecinos y amigos.
- Libertad para mejorar el programa y publicar las mejoras. También exige el código fuente.

1.5.4 Arquitectura de tres capas

La arquitectura de tres capas es un diseño reciente que introduce una capa intermedia en el proceso. Cada capa es un proceso separado y bien definido corriendo en plataformas separadas. En la arquitectura tradicional de tres capas se instala una interfaz de usuario en la computadora del usuario final (el cliente). La arquitectura basada en Web transforma la interfaz de búsqueda existente (el explorador de Web), en la interfaz del usuario final (Truel, 2006).

Con la arquitectura de tres capas, la interfaz del cliente no es requerida para comprender o comunicarse con el receptor de los datos. Por lo tanto, esa estructura de los datos puede ser modificada sin cambiar la interfaz del usuario. El código de la capa intermedia puede ser reutilizado por múltiples aplicaciones si está diseñado en formato modular y gracias a que cada capa es independiente de la otra los cambios o actualizaciones pueden ser realizados sin afectar la aplicación como un todo (Truel, 2006). Los principales beneficios del estilo de arquitectura de tres capas son:

1. Mejoras en las posibilidades de mantenimiento. Debido a que cada capa es independiente de la otra, los cambios o actualizaciones pueden ser realizados sin afectar la aplicación como un todo.

2. Escalabilidad. Como las capas están basadas en diferentes máquinas, el escalamiento de la aplicación hacia afuera es razonablemente sencillo.
3. Flexibilidad. Como cada capa puede ser manejada y escalada de forma independiente, la flexibilidad se incrementa.
4. Disponibilidad. Las aplicaciones pueden aprovechar la arquitectura modular de los sistemas habilitados usando componentes que escalan fácilmente lo que incrementa la disponibilidad (Truel, 2006).

Modelo-Vista-Controlador (MVC)

En el diseño de aplicaciones con sofisticadas interfaces se utiliza el patrón de diseño Modelo-Vista-Controlador (MVC). Pues la lógica de una interfaz de usuario cambia con más frecuencia que los almacenes de datos y la lógica del negocio. Por lo que si se realiza un diseño que mezcle los componentes de interfaz y de negocio, entonces la consecuencia será que, cuando se necesite cambiar la interfaz, se tendrán que modificar los componentes de negocio, siendo esto trabajoso y engorroso, además de provocar posibles errores en el sistema. Por lo que se trata de realizar un diseño que desacople la vista del modelo, con la finalidad de mejorar la reusabilidad. De esta forma las modificaciones en las vistas impactan en menor medida en la lógica de negocio (Reenskaug, 1979).

Elementos del patrón: El **Modelo** es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El Modelo no tiene conocimiento específico de los Controladores o de las Vistas, ni siquiera contiene referencias a ellos (Tanenbaum, 2003).

La **Vista** es el objeto que maneja la presentación visual de los datos representados por el Modelo. Genera una representación visual del Modelo y muestra los datos al usuario (Tanenbaum, 2003). El **Controlador** es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el Modelo. Sirve como un intermediario entre el Modelo, la Vista (Tanenbaum, 2003).

Se trata de realizar un diseño que desacople la vista del modelo, con la finalidad de mejorar la reusabilidad. De esta forma las modificaciones en las vistas impactan en menor medida en la lógica de negocio o de datos. La siguiente figura tomada de (Tanenbaum, 2003) representa de forma sencilla el esquema o patrón MVC:

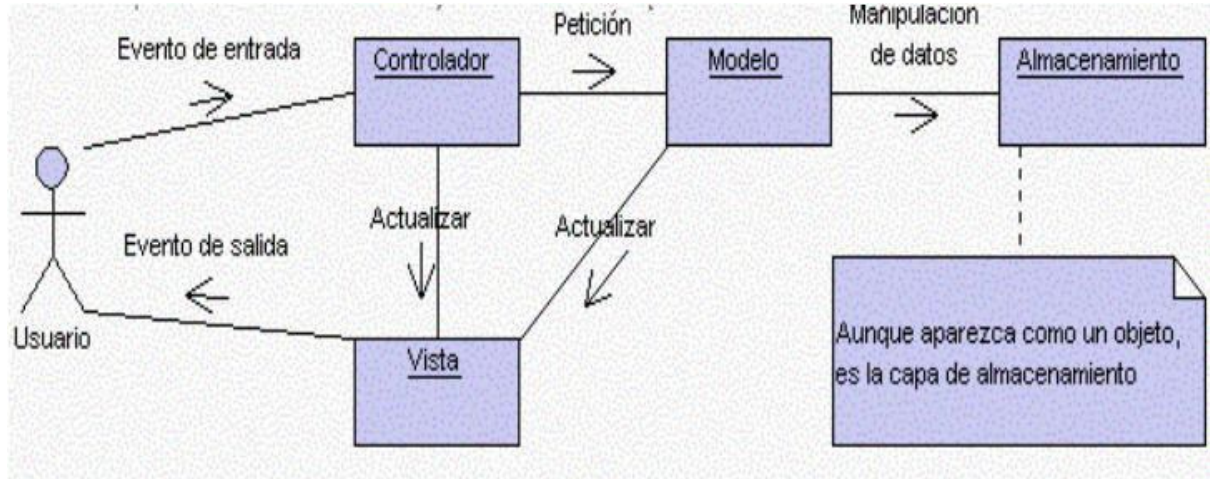


Figura 1 Esquema Modelo - Vista - Controlador.

Todas estas capas pueden residir en un mismo equipo. Lo más usual es que haya una multitud de computadoras donde resida la capa de presentación, es decir los clientes de la arquitectura cliente/servidor, y una computadora central que ejerce la función de servidor en dicha arquitectura. Las capas de negocio y de datos pueden residir en un mismo servidor y si el crecimiento de las necesidades lo aconseja se pueden separar en dos o mas servidores. Así mismo, si el tamaño o la complejidad de la base de datos aumentaran, se puede separar en varios servidores de datos, los cuales recibirán las peticiones del servidor donde reside la capa de negocio. Si por el contrario fuese la complejidad en la capa de negocio lo que obligase a la separación, esta capa de negocio podría residir en una o más computadoras que realizarían solicitudes a una única base de datos. En sistemas muy complejos se llega a tener una serie de ordenadores sobre los cuales corre la capa de datos, y otra serie de ordenadores sobre los cuales corre la base de datos.

Ventajas de este patrón:

Este modelo de arquitectura presenta varias ventajas:

- Hay una clara separación entre los componentes de un programa; lo cual nos permite implementarlos por separado.
- Debido a que cada parte es independiente de la otra los cambios o actualizaciones pueden ser realizados sin afectar la aplicación como un todo.
- La conexión entre el Modelo y sus Vistas es dinámica; se produce en tiempo de ejecución, no en tiempo de compilación (Tanenbaum, 2003).

1.6 Lenguajes de programación

Java:

Java es un lenguaje de programación orientado a objetos desarrollado por la empresa Sun Microsystems en 1995 y que se ha extendido ampliamente en Internet. Es un lenguaje de alto nivel y propósito general, similar a C++, con marcadas características de seguridad y transportabilidad (Camallea & Abalo, 2005).

Uno de los rasgos más importantes es que los programas “ejecutables”, creados por el compilador de Java, son independientes de la arquitectura. Se ejecutan indistintamente en una gran variedad de equipos con diferentes microprocesadores y sistemas operativos. Pero además Java se caracteriza por:

- Ser un lenguaje intrínsecamente orientado a objetos.
- Funcionar perfectamente en red.
- Aprovechar características de la mayoría de los lenguajes modernos evitando sus inconvenientes. En particular los del C++.
- Tener una gran funcionalidad gracias a sus librerías (clases).
- NO tener punteros manejables por el programador, aunque los maneja interna y transparentemente.
- El manejo de la memoria no es un problema, la gestiona el propio lenguaje y no el programador.

- Generar aplicaciones con pocos errores posibles (Schildt, 2005)

Java es uno de los lenguajes de programación más utilizados en el desarrollo de aplicaciones de código abierto y posee una variada bibliografía para aquellos programadores interesados en el aprendizaje de este lenguaje. Por todas estas razones fue tomado en consideración el lenguaje de programación Java para la realización de este proyecto.

1.7 Herramientas de desarrollo

NetBeans:

NetBeans es un entorno de desarrollo integrado (IDE) de gran éxito con una gran base de usuarios, este proyecto de código abierto fue creado por Sun Microsystems en junio de 2000 el cual continúa siendo el patrocinador principal de los proyectos (Netbeans, 2009).

La Plataforma NetBeans es una base modular y extensible usada como una estructura de integración para crear aplicaciones de escritorio grandes. Empresas independientes asociadas, especializadas en desarrollo de software, proporcionan extensiones adicionales que se integran fácilmente en la plataforma y que pueden también utilizarse para desarrollar sus propias herramientas y soluciones (Netbeans, 2009). La plataforma ofrece servicios comunes a las aplicaciones de escritorio, permitiéndole al desarrollador enfocarse en la lógica específica de su aplicación. Entre las características de la plataforma están:

- Administración de las interfaces de usuario (ej. menús y barras de herramientas)
- Administración de las configuraciones del usuario
- Administración del almacenamiento (guardando y cargando cualquier tipo de dato)
- Administración de ventanas
- Framework basado en asistentes (diálogos paso a paso)

El IDE NetBeans es un producto libre y gratuito sin restricciones de uso y de amplias comodidades para el programador (Netbeans, 2009).

Para la realización de este proyecto se utilizó el IDE 6.8 de NetBeans, una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas.

Frameworks

El término Framework o marco de trabajo es usado actualmente en el mundo de la informática para designar una aplicación compilada que permita a través de ella generar otras aplicaciones, puede incluir herramientas de diseño, lenguajes de programación, bibliotecas de clases y compiladores (Camallea & Abalo, 2005).

En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta (Gutiérrez, 2006).

Algunas de las características presentes en casi todos son: abstracción de URL y sesiones (no es necesario manipular directamente las URLs ni las sesiones, el framework se encarga de esto); acceso datos (incluyen las herramientas e interfaces necesarias para integrarse con herramientas de acceso a datos, en bases de datos, etc.); controladores (la mayoría de los frameworks que existen implementa una serie de controladores para gestionar eventos) (Gutiérrez, 2006). Luego del análisis de algunos Framework existentes se tomaron como posibles candidatos Hibernate e Ibatis, ambos cumplen las expectativas para el desarrollo del software.

Ibatis

Ibatis es un framework de código abierto basado en capas desarrollado por Apache Software Foundation, que se ocupa de la capa de persistencia. Puede ser implementado en varios lenguajes de programación como son Java y .NET (CMeadors, Goodin & Clinton).

La mayor parte las diferencias entre Hibernate e Ibatis provienen del hecho de que el último basa su funcionamiento en el mapeo de sentencias SQL que se incluyen en ficheros XML. Eso significa que, al contrario de Hibernate, Ibatis requiere conocimiento de SQL por parte del

programador. Además permite la optimización de las consultas, ya sea con lenguaje estándar o con SQL propietario del motor de base de datos utilizado (CMeadors, Goodin & Clinton).

Al utilizar Ibatis es posible subdividir la capa de Persistencia en tres subcapas (CMeadors, Goodin & Clinton):

- La capa de Abstracción será el interfaz con la capa de la lógica de negocio.
- La capa de Framework de Persistencia será el interfaz con el gestor de Base de Datos.
- La capa de Driver se ocupa de la comunicación con la propia Base de Datos utilizando un Driver específico para la misma.

Hibernate:

Hibernate es un servicio de consultas y persistencia objeto relacional de alto desempeño muy poderoso. Permite desarrollar clases persistentes siguiendo el paradigma de orientación a objetos, soporta relaciones entre objetos de tipo asociación, herencia simple, polimorfismo, composición, y colecciones. Además expresa consultas en su propia extensión portable de SQL (HQL), o en SQL nativo (Bauer & King, 2005). Hibernate, como todas las herramientas de su tipo busca solucionar el problema de la diferencia entre los dos modelos de datos coexistentes en una aplicación: el usado en la memoria de la computadora (orientación a objetos) y el usado en las bases de datos (modelo relacional). Para lograr esto permite al desarrollador detallar cómo es su modelo de datos, qué relaciones existen y qué forma tienen. Con esta información Hibernate le permite a la aplicación manipular los datos de la base operando sobre objetos, con todas las características de la Programación Orientada a Objetos (POO).

Hibernate, como todas las herramientas de su tipo busca solucionar el problema de la diferencia entre los dos modelos de datos coexistentes en una aplicación: el usado en la memoria de la computadora (orientación a objetos) y el usado en las bases de datos (modelo relacional). Para lograr esto permite al desarrollador detallar cómo es su modelo de datos, qué relaciones existen y qué forma tienen. Con esta información Hibernate le permite a la aplicación manipular los datos de la base operando sobre objetos, con todas las características de la Programación Orientada a Objetos (POO).

Hibernate convierte los datos entre los tipos utilizados por Java y los definidos por SQL. Hibernate genera las sentencias SQL y libera al desarrollador del manejo manual de los datos que resultan de la ejecución de dichas sentencias, manteniendo la portabilidad entre todos los motores de bases de datos con un ligero incremento en el tiempo de ejecución (Hibernate, 2009). Entre sus principales características están:

- No intrusivo (estilo POJO)
- Muy buena documentación (forums para ayuda, libro)
- Comunidad activa con muchos usuarios
- Transacciones, caché, asociaciones, polimorfismo, herencia, lazy loading, persistencia transitiva, estrategias de fetching.
- Potente lenguaje de Consulta (HQL)
- Fácil testeo
- No es estándar

El framework seleccionado para el desarrollo de la aplicación fue el Hibernate.

1.8 Sistemas Gestores de Bases de Datos

Una base de datos (BD) es un conjunto de datos interrelacionados entre sí, almacenados con carácter más o menos permanente en la computadora. O sea, que una BD puede considerarse una colección de datos variables en el tiempo (Mato, 2006).

El “software que permite la utilización y/o la actualización de los datos almacenados en una (o varias) base(s) de datos por uno o varios usuarios desde diferentes puntos de vista y a la vez”, se denomina sistema de gestión de bases de datos (SGBD) (Mato, 2006, pág. 10). El objetivo fundamental de un SGBD “consiste en suministrar al usuario las herramientas que le permitan manipular, en términos abstractos, los datos, o sea, de forma que no le sea necesario conocer el modo de almacenamiento de los datos en la computadora, ni el método de acceso empleado” (Mato, 2006, pág. 10). Existen varios SGBD entre los que se encuentran PostgreSQL y SQL Server. PostgreSQL:

PostgreSQL es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) que ha sido desarrollado de varias formas desde la década de 1980. Es un sistema de base de datos relacional libre, liberado bajo la licencia BSD (Berkeley Software Distribution) típico de Unix. Dispone de una serie de funcionalidades que caracterizan a las bases de datos de altas prestaciones que lo hacen apto para la mayoría de las aplicaciones como son (Johnson, 2007):

- Claves ajenas también denominadas llaves ajenas o llaves foráneas (foreignkeys)
- Disparadores (triggers)
- Vistas
- Integridad transaccional
- Acceso concurrente multiversión (no se bloquean las tablas, ni siquiera las filas, cuando un proceso escribe)
- Capacidad de albergar programas en el servidor en varios lenguajes
- Herencia de tablas
- Tipos de datos y operaciones geométricas

Es más avanzado que MySQL, el sistema de base de datos estándar que se emplea en blogs, portales, foros, webs personales, etcétera, aunque es más lento, y sus capacidades no se aprovechan normalmente, por lo que es menos popular que MySQL. Entre las facilidades que brinda PostgreSQL podemos mencionar: la restauración continua de la base de datos, es decir, puedes volver a un punto concreto. Es de suponer que esto representa una carga más para el sistema, pero es una opción interesante. Por otro lado ofrece mejoras de rendimiento y decisiones sobre el sistema de ficheros donde quieres guardar cosas, cambio de tipos de campo con alter table, entre otras. (Johnson, 2007) PostgreSQL, no tiene costo asociado a la licencia del software. Ha sido diseñado y creado para tener un mantenimiento y ajuste mucho menor que otros productos, conservando todas las características, estabilidad y rendimiento. PostgreSQL usa una estrategia de almacenamiento de filas llamada MVCC para conseguir una mejor respuesta en ambientes de grandes volúmenes.

Ventajas

- El costo es la principal ventajas de PostgreSQL.
- La habilidad para poder mirar el código fuente y entender que está sucediendo.
- Permite una fácil gestión de los usuarios y de las bases de datos del sistema.
- PostgreSQL es libre, el software Open Source
- PostgreSQL tiene una legendaria confiabilidad y estabilidad.
- Es muy escalable y extensible.
- Es de plataforma cruzada.
- Está diseñado para entornos de alto volumen.
- PostgreSQL es fácil de administrar.

Desventajas

- Limitaciones al escribir funciones y procedimientos en comparación con Oracle's PL/SQL o Sybase's T-SQL.
- Carencia de herramientas de desarrollo propia.
- La velocidad de respuesta que ofrece este gestor con bases de datos relativamente pequeñas es un poco deficiente.

SQL Server

Diseñada desde su inicio para trabajar en entornos Internet e Intranet, Microsoft SQL Server es capaz de integrar los nuevos desarrollos para estos entornos específicos con los desarrollos heredados de aplicaciones tradicionales (Mato, 2006). SQL soporta la configuración automática y la auto-optimización, además tiene una administración multiservidor para un gran número de servidores, pero es mayormente utilizada para trabajar en Plataformas de Windows. (Mato, 2006). Características:

- Almacenamiento de datos.
- Soporte de transacciones.
- Escalabilidad, estabilidad y seguridad.
- Soporta procedimientos almacenados.
- Incluye también un potente entorno gráfico de administración
- Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo acceden a la información.
- Permite administrar información de otros servidores de datos.
- Facilidad de instalación, distribución y utilización.

Después de analizar los diferentes SGBD se selecciona el PostgreSQL como la herramienta óptima para la base de datos.

Conclusiones

En este capítulo se realizó una revisión bibliografía para conocer los conceptos fundamentales de la investigación, se estudiaron las principales herramientas para elaborar el software y atendiendo a sus características se escogieron las siguientes:

- RUP (Proceso Unificado de Desarrollo) como la metodología más apropiada para el desarrollo del proyecto.
- UML como el lenguaje de modelación utilizado.
- La arquitectura Cliente-Servidor en tres Capas.
- Para la implementación el lenguaje Java, utilizando como herramienta de programación NetBeans IDE 6.8 y el framework Hibernate.
- PostgreSQL como sistema gestor de bases de datos
- Rational Rose como herramienta de modelado UML.

La selección de estas tecnologías se basó principalmente en la tecnología código libre y la característica de integrarse para funcionar en cualquier plataforma.

Capítulo II: Diseño de un software que contribuya a la gestión de información de la Empresa de Transporte de Sancti Spíritus.

2.1 Introducción

En este capítulo se realiza un estudio del modelo del negocio, este proceso permite una mejor comprensión de la problemática al conocer cuales son los actores del negocio, los trabajadores, los casos de uso, entre otros. Además se realizan diagramas de actividades, modelos de objeto y expansiones a los casos de uso, los cuales esclarecen el proceso a automatizar y los pasos que no se deben obviar.

Además en este capítulo se dejan plasmados los casos de uso del sistema, sus actores, diagramas de casos de uso del sistema así como la descripción de cada uno según lo propuesto por la metodología RUP.

2.2 Modelo del negocio.

El modelado de negocio es una técnica para comprender los procesos de negocio de la organización. Además de identificar los casos de uso y las entidades del negocio relevantes que el software debe soportar, de forma que se puede modelar solo lo necesario para que se comprenda el contexto (Jacobson, Booch & Rumbaugh, 2000).

El modelo del negocio está soportado por dos tipos de modelos de UML: modelo de casos de uso y modelo de objetos. (Jacobson, Booch & Rumbaugh, 2000)

A continuación describimos el proceso de negocio que se lleva a cabo en la Empresa de Transporte de la Construcción de Sancti Spíritus, mediante los artefactos propuestos por la metodología RUP y modelados por el lenguaje UML.

2.1.1 Identificación de los procesos del negocio

Cuando se hable de procesos de negocio se puede decir que son un grupo de tareas relacionadas lógicamente que se llevan a cabo en una determinada secuencia y forma, y que emplean los recursos de la organización para dar resultados que apoyen sus objetivos (Rumbaugh, Booch & Jacobson, 2006).

A partir del proceso planteado anteriormente se identificó el siguiente proceso de negocio:

La Empresa de Transporte de la Construcción de Sancti Spíritus inicia su procedimiento de trabajo a través de la demanda de materiales de la construcción que emite cada obra y a partir del cual la empresa conforma su plan de transportación del mes. A partir de ahí comienza la transportación de la forma siguiente:

Cada viaje ha de estar avalado por un modelo denominado Carta de Porte que acompañará al chofer durante todo el recorrido desde su salida de la Empresa, llegada, espera y carga en el centro de carga; regreso llegada, espera y descarga en el centro de descarga y vuelta a la empresa donde entrega dicha carta de Porte en la que se ha ido recogiendo información en los diferentes puntos del recorrido. A partir de esta última la dirección de la empresa conoce e informa algunos parámetros de interés como producción en físico, en valores, consumo de combustible, y varios más a diferentes instancias que lo solicitan (GECSS, Partido, Gobierno, etc.) por lo que los técnicos de producción y el técnico de recursos humanos recogen estos datos para introducirlos en el sistema.

2.1.2 Reglas del negocio

Las reglas del negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto de negocio (Rumbaugh, Booch & Jacobson, 2006).

- El chofer no debe salir de la entidad sin tener en su poder la planilla de la Carta de Porte.
- No puede aceptar descargar o cargar ninguna mercancía sin esta planilla.
- Para realizar algún contrato de servicio se debe traer un documento firmado por la empresa interesada en recibir el servicio de transportación.

- Los técnicos de producción no deben aceptar la planilla de la Carta de Porte incompleta al igual que el técnico de recursos humanos.

2.1.3 Modelo de casos de uso del negocio

El modelo de Casos de Uso del Negocio es el encargado de describir los procesos de una empresa utilizando los casos de uso y los actores, en correspondencia, a su vez, con los procesos del negocio y los clientes (Jacobson, Booch, & Rumbaugh, 2000). Se define a través de: el diagrama de casos de uso, la descripción y el diagrama de actividades de los casos de uso.

2.1.4 Actores del negocio

Se considera actor del negocio a cualquier individuo, grupo, entidad, organización, máquina o sistema de información externo; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados (Jacobson, Booch & Rumbaugh, 2000).

Actores del negocio	Justificación
Cliente	Interesado en contratar los servicios que presta la empresa.
Director	Responsable del buen funcionamiento de la empresa por lo que debe recibir reportes actualizados del proceso productivo.
Chofer	Interesado en dar los viajes programados por la empresa

Tabla 1 Actores del negocio

2.1.5 Diagrama de casos de uso del negocio

Los diagramas de casos de uso se utilizan para especificar las funcionalidades y el comportamiento de un sistema mediante su interacción con los usuarios y otros sistemas. O sea es un diagrama que muestra la relación entre los actores y los casos de uso en un sistema (Jacobson, Booch, & Rumbaugh, 2000).

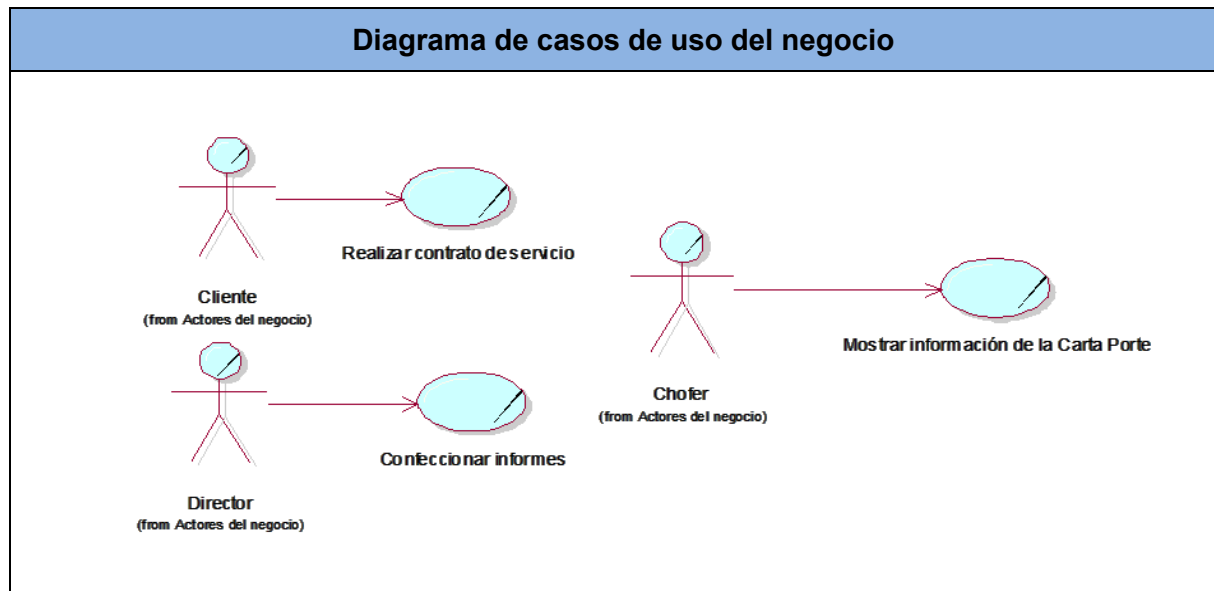


Figura 2 Diagrama de casos de uso del negocio

2.1.6 Trabajadores del negocio

Un trabajador es una abstracción de una persona o grupo de personas, una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio, manipulando entidades del mismo y representando un rol (González, 2005).

Los trabajadores identificados son los siguientes:

Trabajadores del negocio	Justificación
Técnico de producción 1	Controla la producción diaria de la empresa.
Técnico de producción 2	Controla el consumo de combustible diario.
Técnico de Recursos Humanos	Realiza el informe de pago.

Tabla 2 Trabajadores del negocio

2.1.7 Descripción de los casos de uso del negocio

CU1 Realizar contrato de servicio	
Actores del negocio	Clientes
Propósito	Controlar cada uno de los clientes que han solicitado los servicios de la empresa.
Resumen: Comienza cuando una entidad viene a la ETC a solicitar los servicios de transportación de sus mercancías. Se toman sus datos y se le hace un contrato de prestación de servicios para de esa manera poder sacar a través de facturas la forma de pago a la empresa.	
Flujo de trabajo	
Acción del actor	Respuesta del negocio
1- Comienza con el cliente (en este caso alguna entidad estatal) solicita por escrito el servicio de transportación de mercancía a través de la ETC.	2- Se revisan sus datos, si están en orden se registran, de lo contrario ir a Subsección 1.
	3- Se le confecciona un contrato de servicio.
	4- Se le entrega una copia de contrato.
5- El cliente recibe la copia y así recibe la prestación de servicio.	
Curso Alternativo de los eventos	
Subsección 1	6- Se le informa que tiene datos desactualizados.
7- El cliente recibe la información.	
Prioridad:	Alta

Tabla 3 Descripción del caso de uso: Realizar contrato de servicio.

CU2 Confeccionar informes	
Actores del negocio	Director
Propósito	Tener conocimiento de las acciones realizadas por la ETC y el nivel de producción en que medida se encuentra.
Resumen: Comienza cuando el director solicita a los trabajadores un informe completo con toda la información referida a la ETC, estos realizan el informe y se lo entregan.	
Flujo de trabajo	
Acción del actor	Respuesta del negocio
1- Comienza cuando el director solicita los informes.	
	2- Los trabajadores realizan los informes y se lo envían.
3- El director recibe los informes.	
Prioridad:	Baja
Mejoras	Al estar digitalizada toda la información que se maneja en la empresa la elaboración de los reportes tomará menor cantidad de tiempo y ofrecerá una mayor confiabilidad en los datos.

Tabla 4 Descripción del caso de uso: Confeccionar informes.

CU3 Llenar información de la Carta de Porte	
Actores del negocio	Chofer
Propósito	Dar a conocer e informar a la empresa de algunos parámetros de interés como producción en físico, en valores, consumo de combustible, y varios más a diferentes instancias que lo soliciten.
Resumen: Comienza cuando el chofer sale de la empresa y empieza su recorrido en llegada, espera y carga en el centro de carga; regreso llegada, espera y descarga en el centro de descarga y vuelta a la empresa donde entrega dicha carta de Porte en la que se ha ido recogiendo información en los diferentes puntos del recorrido.	
Flujo de trabajo	
Acción del actor	Respuesta del negocio
1- Comienza cuando el chofer toma la Carta de Porte.	
2- Sale de la empresa para comenzar .	

el recorrido por los diferentes puntos.	
3- Llega al centro de carga.	4- Allí lo reciben y le dan los materiales.
5- Este recibe los materiales y los carga.	
6- Solicita los datos para llenar la Carta de Porte.	7- Le dan los datos solicitados.
8- Este los recibe y los coloca en la Carta de Porte.	
9- Luego se dirige al centro de descarga y entrega los materiales.	10- Allí lo reciben y toman estos materiales.
11- Este solicita información para el llenado de la Carta de Porte.	12- Le dan los datos solicitados.
13- Este los recibe y los coloca en la Carta de Porte.	
14-Regresa a la Empresa y entrega dicha Carta de Porte.	
Prioridad:	Alta
Mejoras	Al estar digitalizada toda esta información sobre la producción de la empresa los distintos departamentos trabajarán con una buena calidad y en menor cantidad de tiempo y los informes serán elaborados de forma rápida y ofrecerán una mayor confiabilidad en sus datos.

Tabla 5 Descripción del caso de uso: Llenar información de la Carta de Porte

2.1.8 Diagramas de actividades

Un diagrama de actividades es un diagrama que muestra el flujo de actividad a actividad; los diagramas de actividad tratan la vista dinámica de un sistema. Un caso especial de diagrama de estados (aquellos diagramas que tratan la vista dinámica de un sistema) en el cual todos o casi todos los estados son estados de acción y en el cual todas o casi todas las transiciones son disparadas por la terminación de las acciones en los estados origen (Rumbaugh, Booch & Jacobson, 2006).

Diagrama de actividades: Realizar contrato de servicio

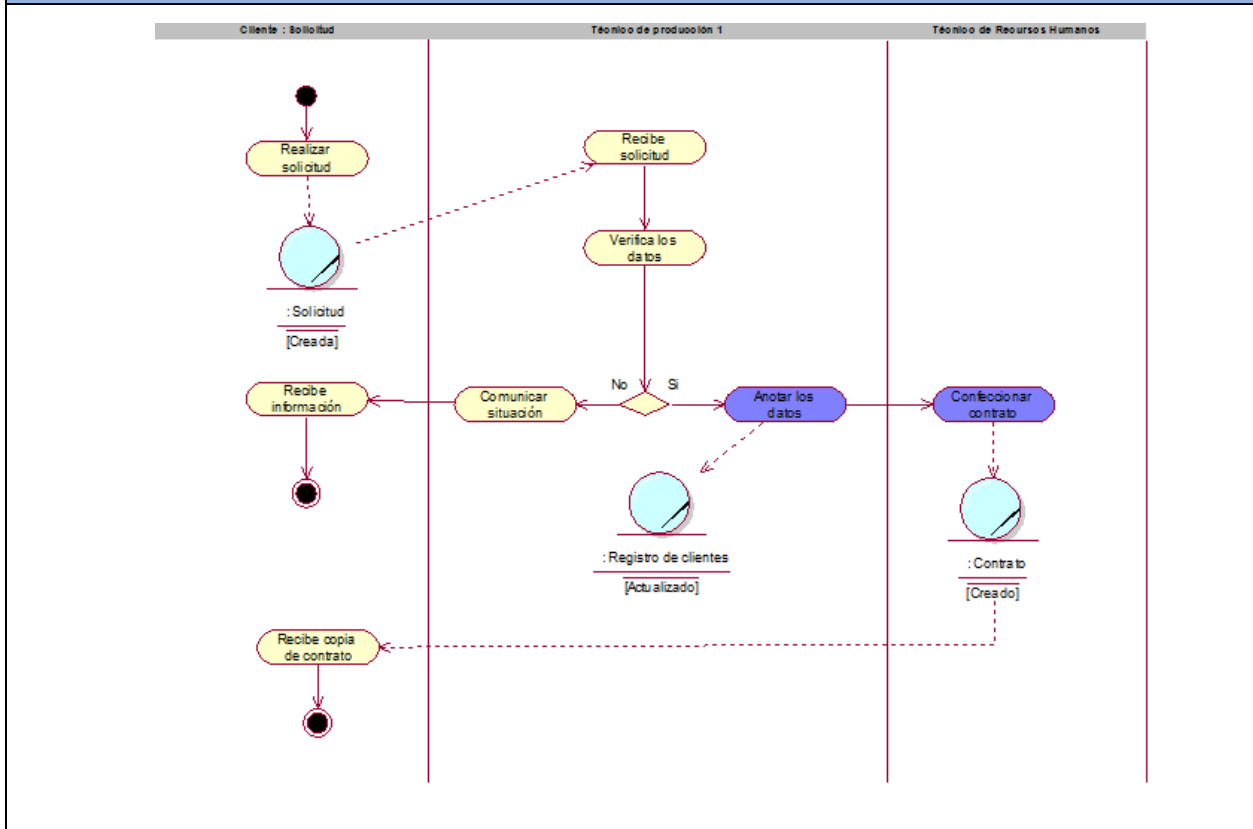


Figura 3 Diagrama de actividades: Realizar contrato de servicio

Diagrama de actividades: Llenar información de la Carta de Porte

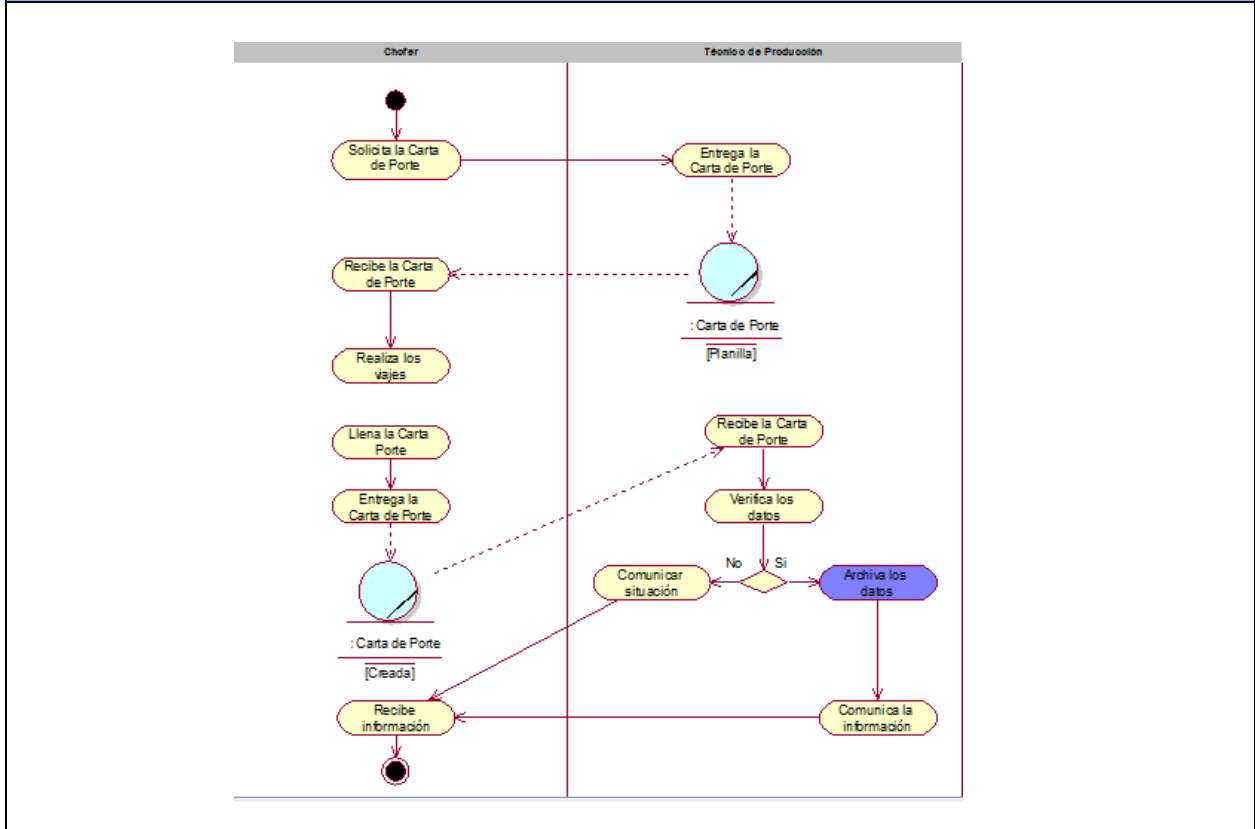


Figura 4 Diagrama de actividades: Llenar información de la Carta de Porte

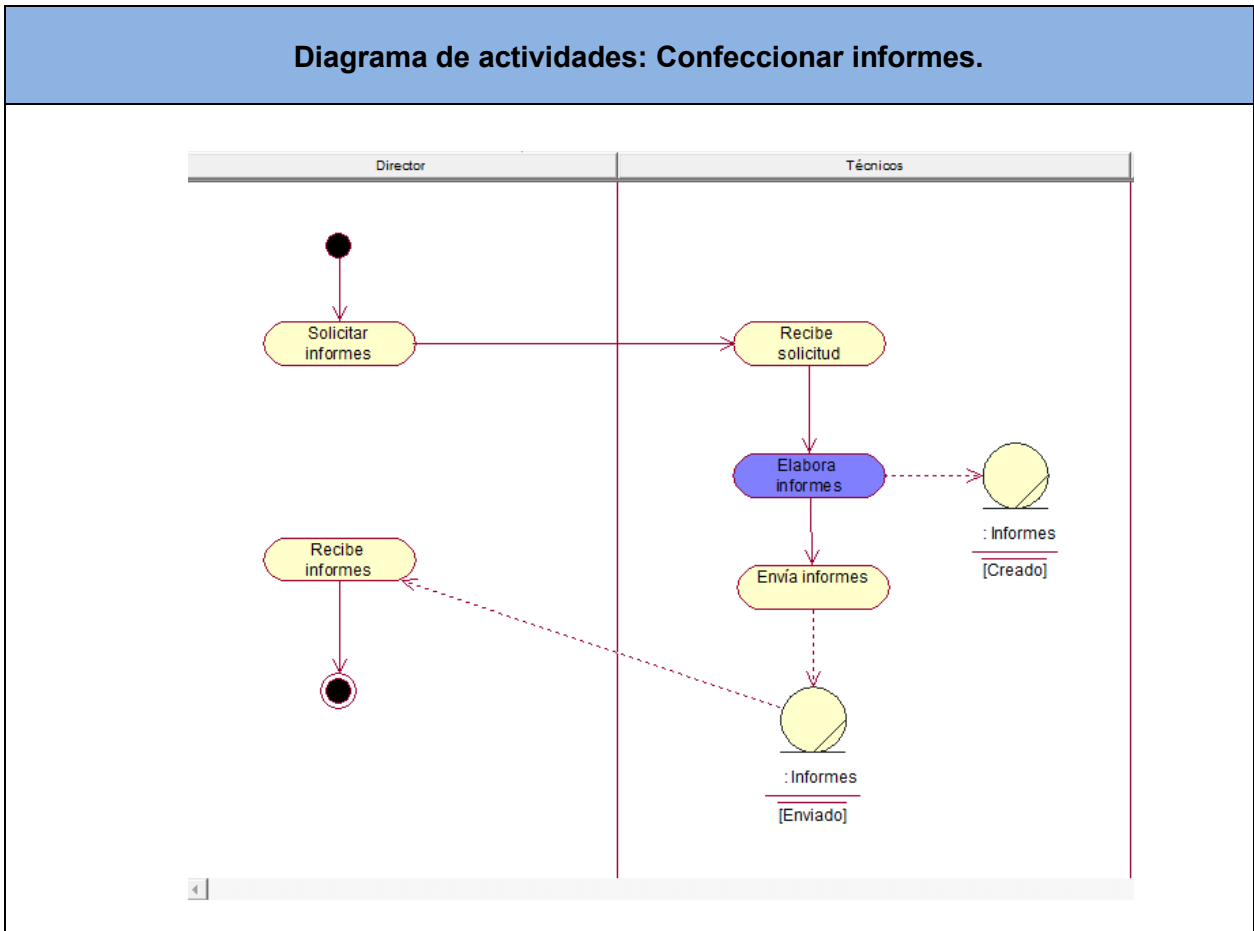


Figura 5 Diagrama de actividades: Confeccionar informes.

2.1.9 Modelo de objetos del negocio

El modelo de objetos del negocio se utiliza para describir la participación de los trabajadores y entidades del negocio, y su colaboración en la realización del negocio. Un modelo de objetos del negocio es un modelo interno a un negocio. Describe como cada caso de uso del negocio es llevado a cabo por parte de un conjunto de trabajadores que utilizan un conjunto de entidades del negocio y unidades de trabajo (González, 2005).

Una entidad del negocio es algo que los trabajadores toman, inspeccionan, manipulan o producen en un caso de uso del negocio (González, 2005). Ver Anexo 2.

2.2 Requerimientos

2.2.1 Requerimientos funcionales

González (2005) plantea que un requerimiento funcional expresa una especificación más detallada de las responsabilidades del sistema que se propone. Ellos permiten determinar, de una manera clara lo que debe hacer el sistema, siempre basándose en las necesidades de los usuarios.

Los requerimientos funcionales del sistema propuesto son:

1. Autenticar usuario
2. Cambiar contraseña
3. Insertar usuario
4. Modificar usuario
5. Eliminar usuario
6. Insertar rol
7. Modificar rol
8. Eliminar rol
9. Insertar municipio
10. Modificar municipio
11. Eliminar municipio
12. Insertar empresa
13. Modificar empresa
14. Eliminar empresa
15. Insertar equipo

16. Modificar equipo
17. Eliminar equipo
18. Insertar origen
19. Modificar origen
20. Eliminar origen
21. Insertar destino
22. Modificar destino
23. Eliminar destino
24. Insertar producto
25. Modificar producto
26. Eliminar producto
27. Insertar parte diario
28. Modificar parte diario
29. Eliminar parte diario
30. Reporte de listado de choferes
31. Reporte de parte diario
32. Reporte de productos transportados al MICONS
33. Reporte de productos transportados por obra
34. Reporte de actividades por tipo de carga y programa
35. Reporte por línea de equipo
36. Reporte de resumen por chofer
37. Reporte de producción por chofer

2.2.2 Requerimientos no funcionales

Requerimientos de Apariencia o Interfaz Externa

- La interfaz estará diseñada de modo tal que el usuario pueda tener en todo momento el control de la aplicación, lo que le permitirá ir de un punto a otro dentro de ella con gran facilidad. Se cuidará que la aplicación sea lo más interactiva posible.

Requerimientos de Usabilidad

- El sistema es de libre acceso, pero a la aplicación diseñada para la actualización de los datos del proceso productivo y la gestión de usuarios solo tendrán acceso los usuarios definidos por el administrador del sistema.

Requerimientos de Rendimiento

- La eficiencia del sistema estará determinada por el aprovechamiento de los recursos que se disponen en la arquitectura 3 capas, y la velocidad de las consultas en la Base de Datos.

Requerimientos de Soporte

- Se requiere un servidor de bases de datos con soporte para grandes volúmenes de información, velocidad de procesamiento y tiempo de respuesta rápido en accesos concurrentes. Se documentará la aplicación para garantizar su soporte.
- Las pruebas del sistema se realizarán en la oficina de producción de la ETC. Dichas pruebas permitirán evaluar en la práctica la funcionalidad y las ventajas de este nuevo producto.

Requerimientos de Portabilidad

- La plataforma seleccionada para desarrollar la aplicación fue Windows.

Requerimientos de Seguridad

- Debe garantizar la conectividad e integridad de los datos almacenados a través de la red. Esto está garantizado por el Sistema Operativo.
- Debe garantizar la confidencialidad para proteger la información de acceso no autorizado. Esto estará garantizado por el Sistema Gestor de Base de Datos.

- El sistema impondrá un estricto control de acceso que permitirá a cada usuario tener disponible solamente las opciones relacionadas con su actividad.
- La información deberá estar disponible a los usuarios en todo momento, limitada solamente por las restricciones que estos tengan de acuerdo con la política de seguridad del sistema.

Confiabilidad

- Garantía de un tratamiento adecuado de las excepciones y validación de las entradas del usuario.
- El sistema en casos de fallos debe garantizar que las pérdidas de información sean mínimas.

Requerimientos de Ayuda y Documentación en Línea

- El sistema contará con una ayuda que explicará de manera clara y sencilla al usuario todas las funcionalidades del sistema.
- La ayuda quedará conformada por un menú general que le facilitará al usuario poder ir de un lugar a otro sin perderse.

Requerimientos de Software

- Se debe disponer de un sistema operativo compatible. El sistema debe poderse ejecutar en entornos Windows y Linux.
- Las máquinas de los clientes deben estar conectadas a una máquina que contenga un servidor de PostgreSQL.

Requerimientos de Hardware

Para la puesta en práctica del proyecto se requieren máquinas con los siguientes requisitos:

- Se requiere de una máquina que funcione como servidor, la cual debe tener como mínimo:
 - ✓ Un Procesador Pentium III con 1 GHz de frecuencia o superior
 - ✓ 512 Mb de Memoria RAM
 - ✓ 40 GB de Disco Duro

Las computadoras situadas en los puestos de trabajo de los usuarios requerirán como mínimo:

- ✓ Un Procesador Pentium III
- ✓ 128 Mb de Memoria RAM
- ✓ Deben estar conectadas en red con el servidor a través de una tarjeta de red de 100 Mbps

Restricciones en el diseño y la implementación

- Se utilizarán herramientas de desarrollo que garanticen la calidad de todo el ciclo de desarrollo del producto.

2.3 Modelo del sistema

2.3.1 Modelo de casos de uso del sistema

El modelado de Casos de Uso es la técnica más simple que emplean los desarrolladores de software para modelar los requisitos del sistema desde la perspectiva del usuario. El modelo de casos de uso consiste en actores y casos de uso. Los actores representan usuarios y otros sistemas que interactúan con el sistema y los casos de uso representan el comportamiento del sistema, los escenarios que el sistema atraviesa en respuesta a un estímulo desde un actor (Popkin Software and System, 2005).

2.3.2 Actores del sistema

Un actor no es más que un rol que juega un usuario de Caso de Uso cuando interactúan con estos casos de uso. Los actores representan a terceros fuera del sistema que colaboran con el mismo. Una vez que se han identificado los actores del sistema, se ha identificado el entorno externo del sistema (Rumbaugh, Booch & Jacobson, 2006).

Teniendo en cuenta lo planteado anteriormente se definieron los siguientes actores del sistema:

Actores del sistema	Descripción
Administrador	Interesado en administrar y controlar el acceso al software.
Técnico	Responsable de introducir cada uno de los datos referido a la producción de la empresa.
Invitado	Interesado en conocer diferentes reportes facilitados por el sistema.

Tabla 6 Actores del sistema.

2.3.3 Casos de uso del sistema

Cada forma en que los actores usan el sistema se representa con un Caso de Uso. Los Casos de Uso son “fragmentos” de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. Un Caso de Uso especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia (Jacobson, Booch, & Rumbaugh, 2006). Dado el número de casos de uso, se introducen paquetes para lograr una mejor comprensión y organización de los elementos en grupos. Sobre los paquetes del sistema Pressman expresa: Subdividir los casos de uso en paquetes resulta de mucha ayuda en la modelación de cualquier sistema informático. Los paquetes son un mecanismo de organización de elementos que subdividen el modelo en otros más pequeños que colaboran entre sí (Pressman, 2007).

Para este sistema se definieron los siguientes casos de uso:

1. Paquete Seguridad

- 1.1 Autenticar usuario
- 1.2 Cambiar contraseña

2. Paquete Catálogo

- 2.1 Gestionar chofer
- 2.2 Gestionar empresa
- 2.3 Gestionar municipio
- 2.4 Gestionar destino
- 2.5 Gestionar origen
- 2.6 Gestionar equipo
- 2.7 Gestionar producto
- 2.8 Gestionar parte diario

3. Paquete Administración

3.1 Gestionar usuarios

3.2 Gestionar roles

4. Paquete Reportes

4.1 Reporte de listado de choferes

4.2 Reporte de parte diario

4.3 Reporte de productos transportados al MICONS

4.4 Reporte de productos transportados por obra

4.5 Reporte de actividades por tipo de carga y programa

4.6 Reporte por línea de equipo

4.7 Reporte de resumen por chofer

4.8 Reporte de producción por chofer

2.3.4 Diagramas de casos de uso del sistema

Para facilitar el trabajo con los casos de uso y la organización de los elementos se agrupan los casos de uso en paquetes, los cuales se muestran a continuación.

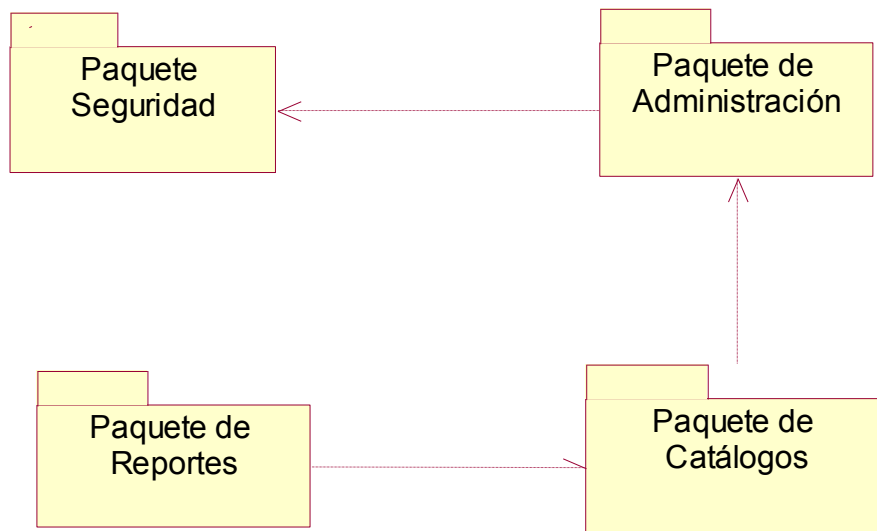


Figura 6 Diagrama de casos de uso por paquetes

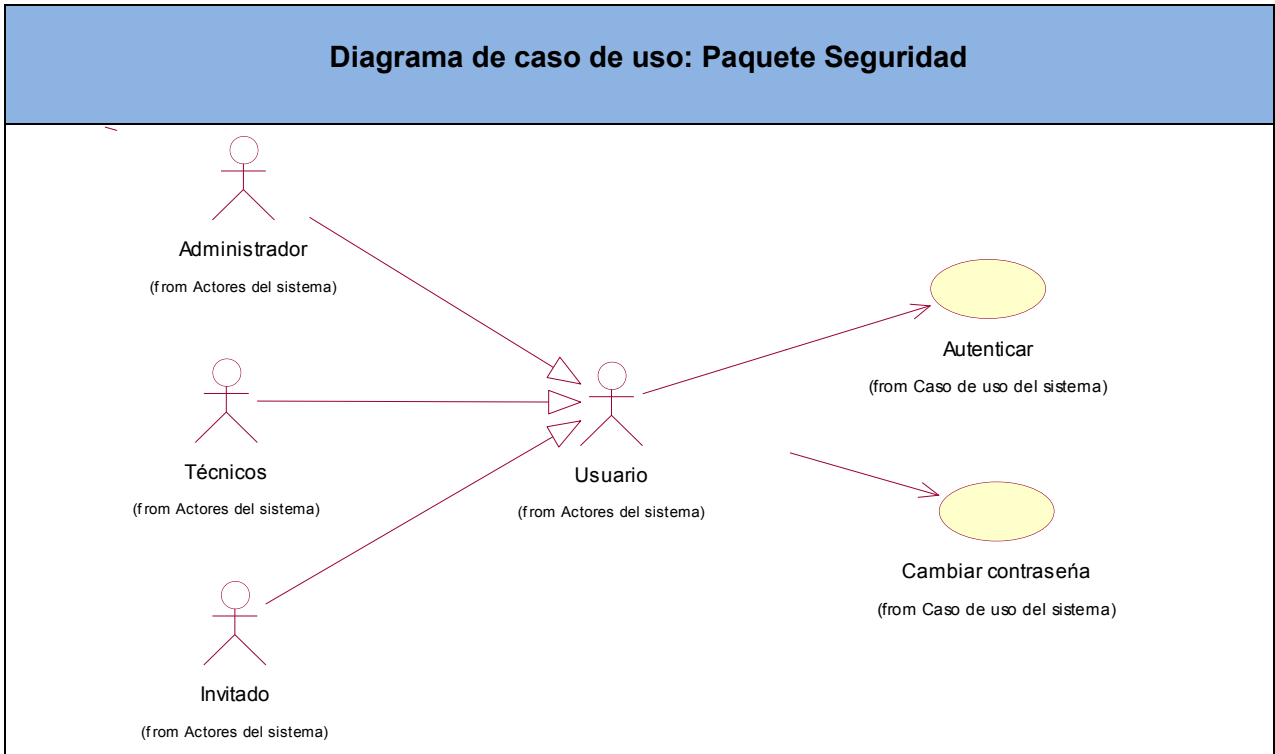


Figura 7 Diagrama de casos de uso: Paquete Seguridad

Descripción de los casos de uso Paquete Seguridad ver [Anexo 2.1](#).

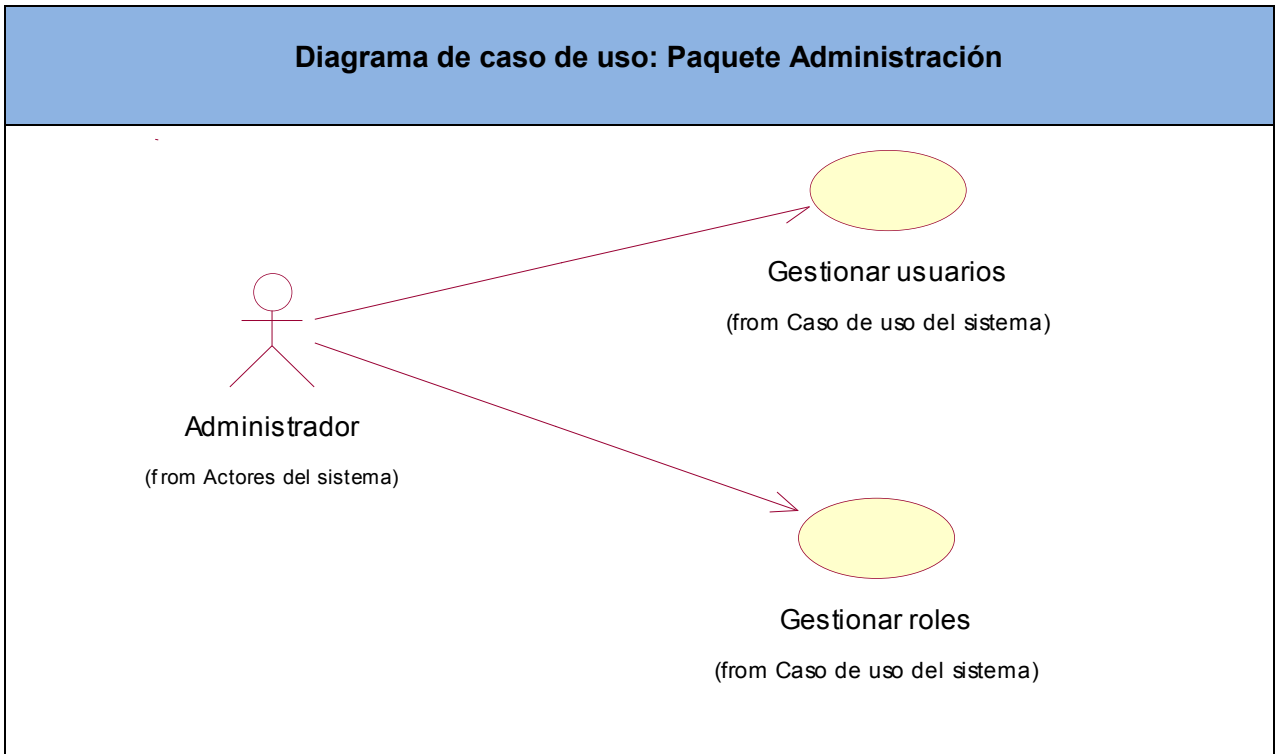


Figura 8 Diagrama de casos de uso: Paquete Administración

Descripción de los casos de uso Paquete Administración ver [Anexo 2.2](#).

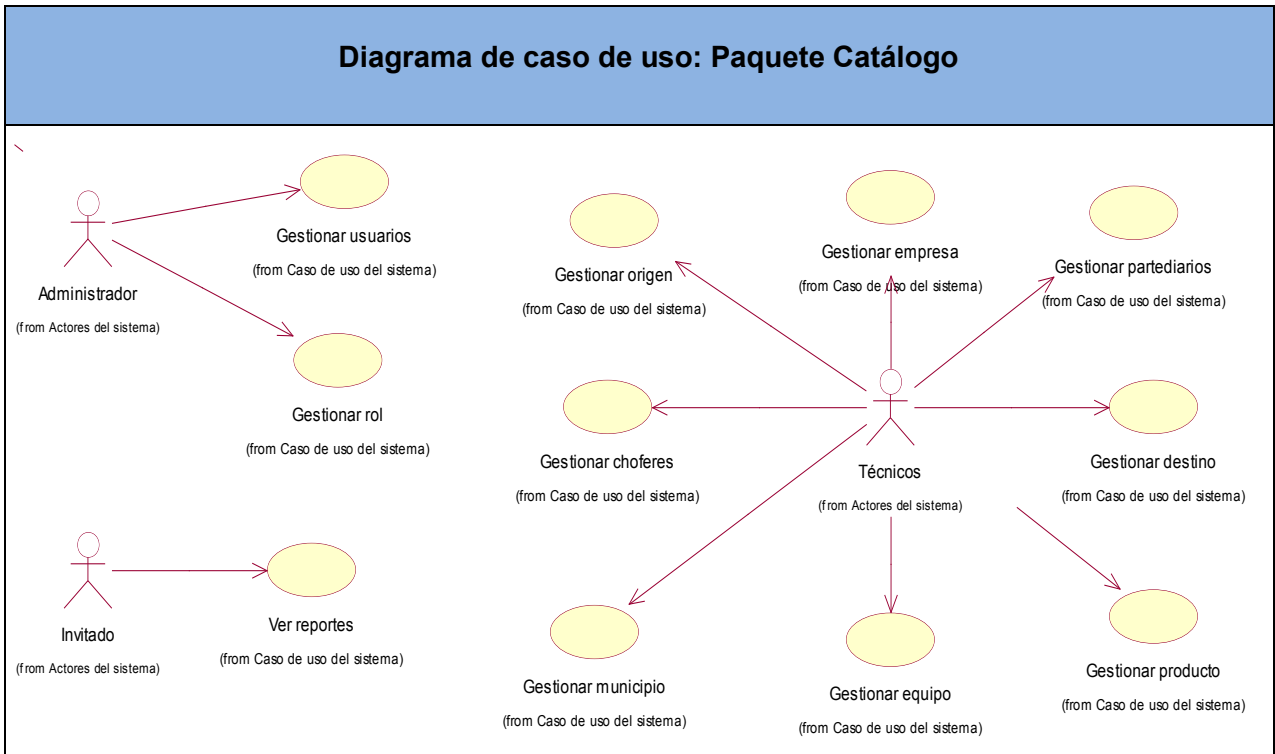


Figura 9 Diagrama de casos de uso: Paquete Catálogo
 Descripción de los casos de uso Paquete Catálogo ver [Anexo 2.3](#).

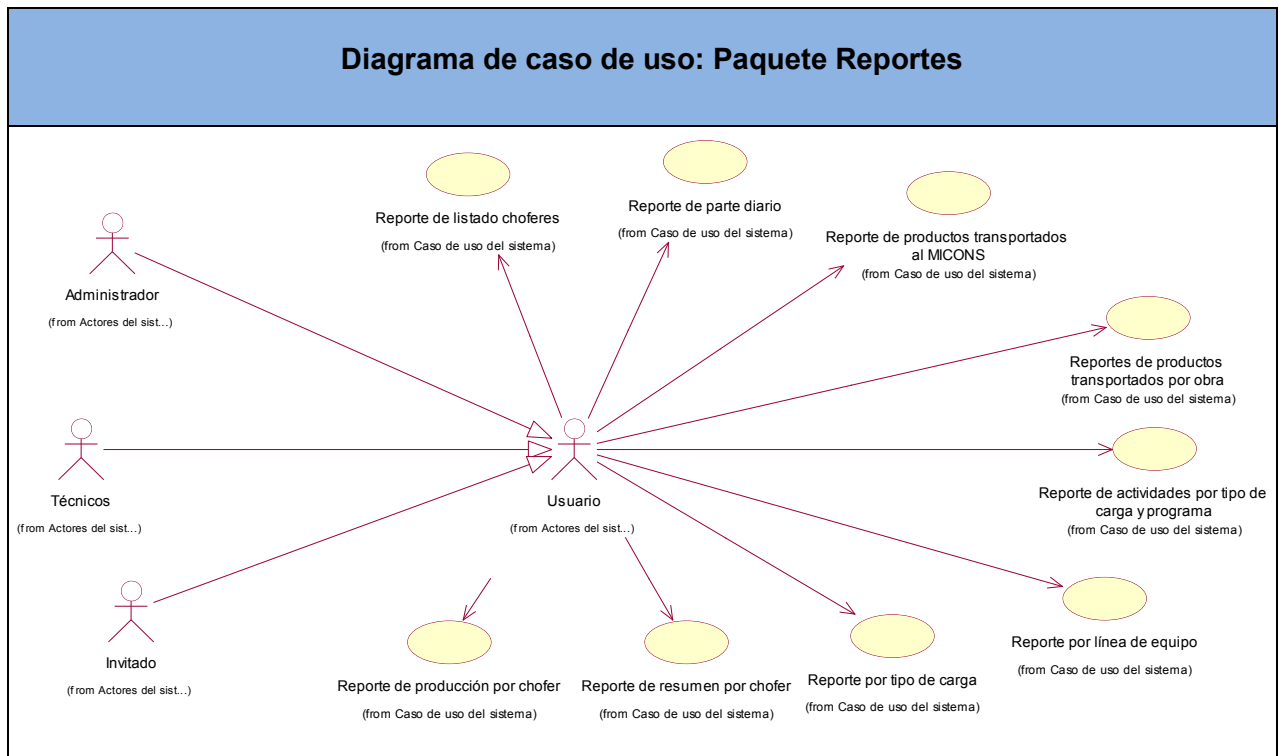


Figura 10 Diagrama de casos de uso: Paquete Reportes

Descripción de los casos de uso Paquete Reportes ver [Anexo 2.4](#).

Conclusiones

Este capítulo deja claro cómo funciona el negocio a través de los artefactos proporcionados por la metodología RUP y las reglas del negocio, entre otros. Además contiene una descripción general del sistema identificando los requerimientos funcionales y no funcionales y los casos de uso. La construcción de todos estos artefactos propició que:

- Se esclareciera cómo es el flujo de eventos que se realiza en cada uno de los casos de uso.
- Se establecieron las relaciones de cada uno de los actores del sistema con los casos de uso.

Capítulo III: Implementación de un software que contribuya a la gestión de información de la Empresa de Transporte de Sancti Spíritus.

Introducción

El presente capítulo abarca todos los aspectos referentes al diseño de la solución propuesta. Durante esta etapa se toman decisiones importantes que conllevan al efectivo cumplimiento de los requerimientos funcionales y la obtención de un software con calidad.

Se plasman los resultados de la etapa del diseño. Se presentan diagramas como por ejemplo: el modelo de datos a partir del diagrama de clases persistentes, el diagrama de componente y de despliegue, que resultaron del diseño realizado de cada uno de los casos de uso del sistema planteados y por último, se hará referencia a los estándares de diseño y de programación que se tienen en cuenta.

3.1 Diagrama de clases del diseño

Una clase de diseño es una abstracción de una clase o construcción similar en la implementación del sistema (Jacobson, Booch, & Rumbaugh, 2000).

Un diagrama de clases es una colección de elementos declaratorios del modelo, como clases, tipos y sus relaciones; conectados unos a otros y a sus contenidos en forma de grafo (Jacobson, Booch & Rumbaugh, 2000).

Los diagramas de clases del diseño fueron elaborados a partir de los diferentes casos de uso del sistema.

Diagrama de clases del diseño Autenticar usuario

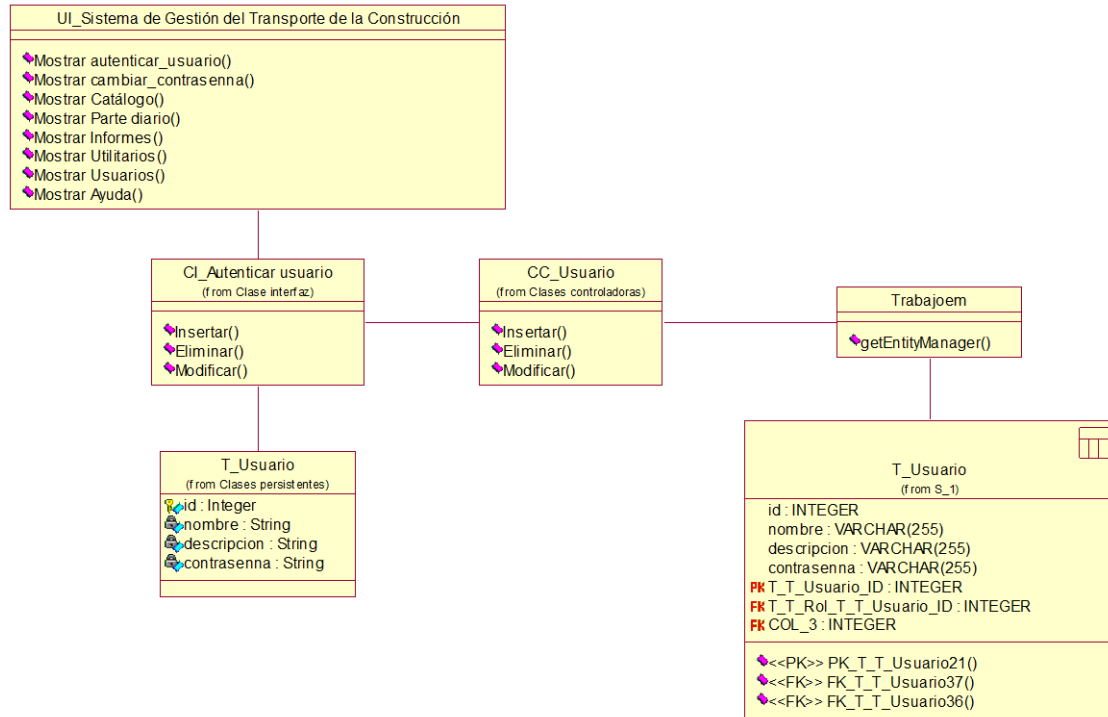


Figura 11 Diagrama de clases del diseño Autenticar usuario

Diagrama de clases del diseño Cambiar contraseña

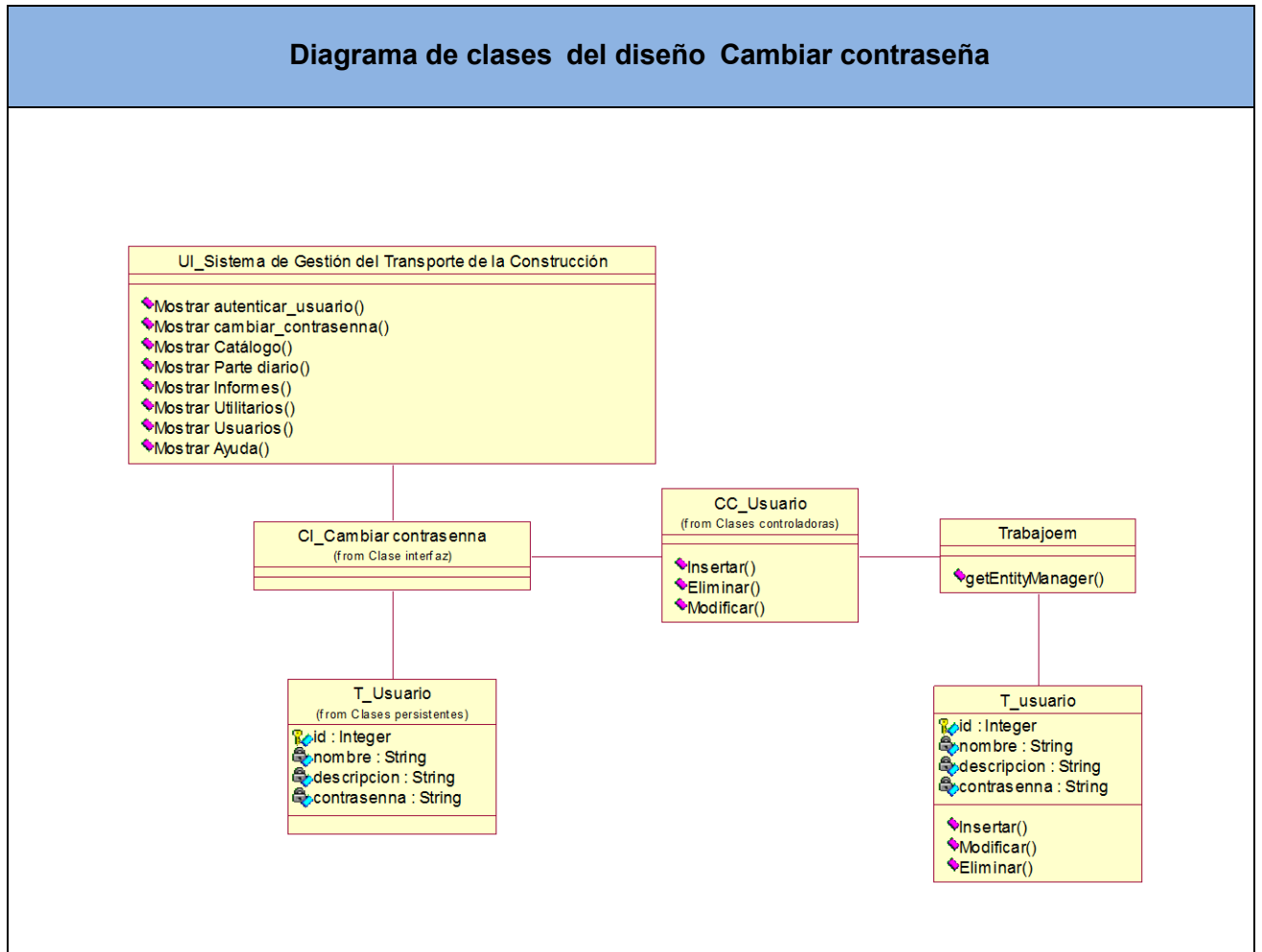


Figura 12 Diagrama de clases del diseño Cambiar contraseña

Diagrama de clases del diseño Parte diario

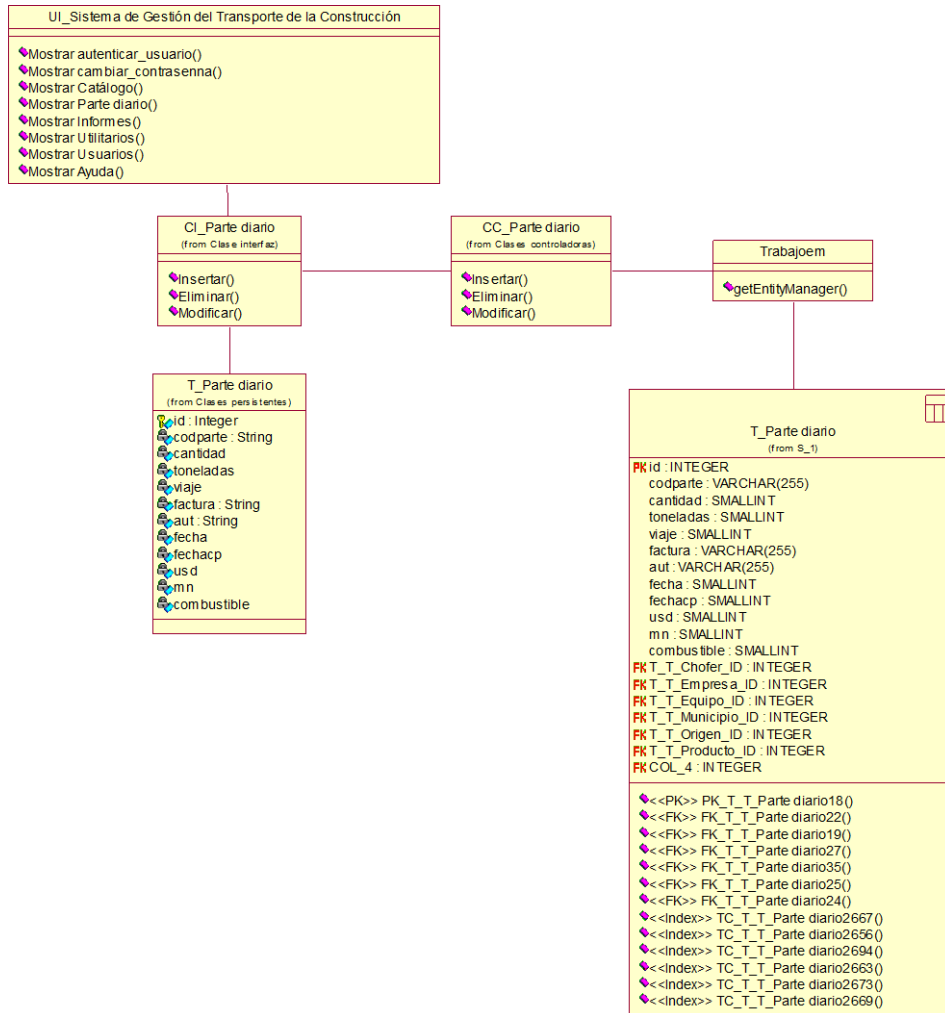


Figura 13 Diagrama de clases del diseño Parte diario

Diagrama de clases del diseño Gestionar chofer

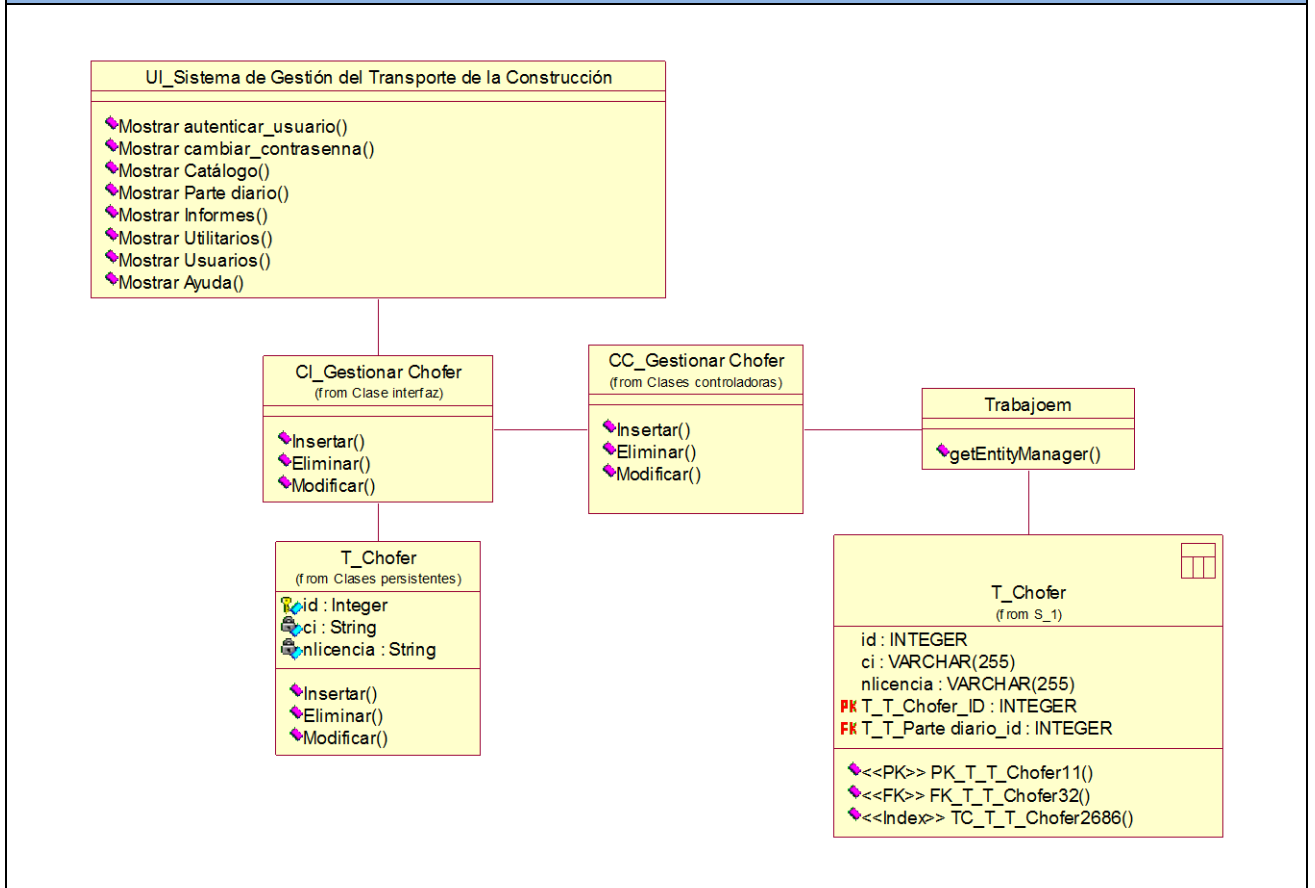


Figura 14 Diagrama de clases del diseño Gestionar chofer

Diagrama de clases del diseño Gestionar destino

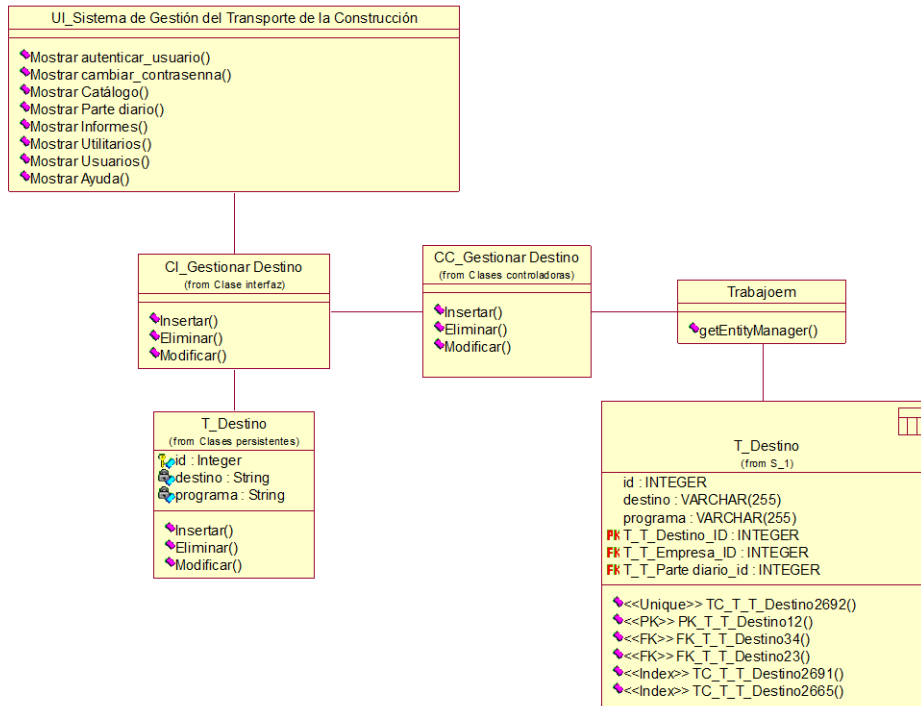


Figura 15 Diagrama de clases del diseño Gestionar destino

Diagrama de clases del diseño Gestionar empresa

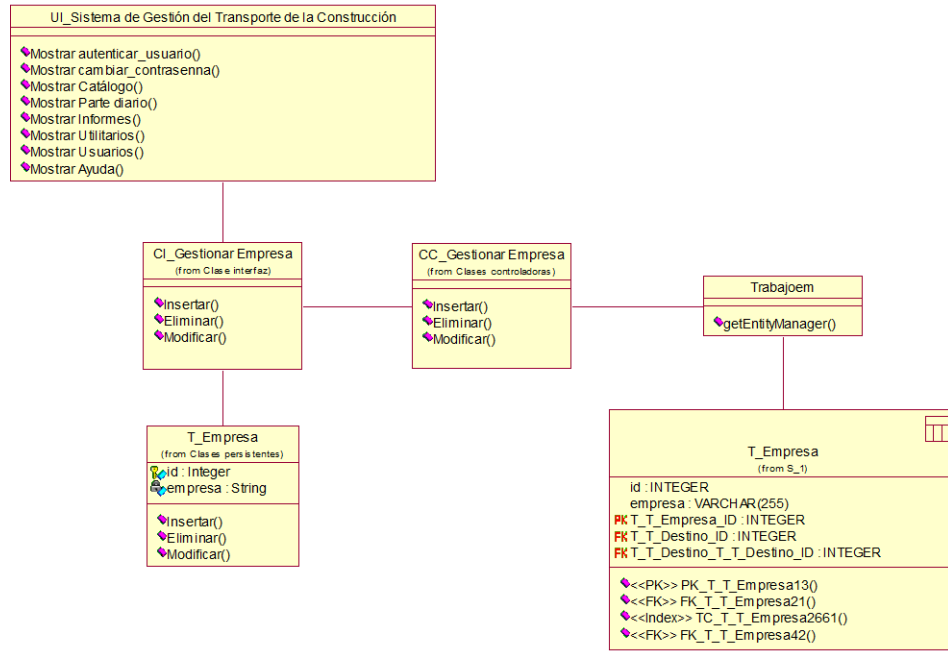


Figura 16 Diagrama de clases del diseño Gestionar empresa

Diagrama de clases del diseño Gestionar equipo

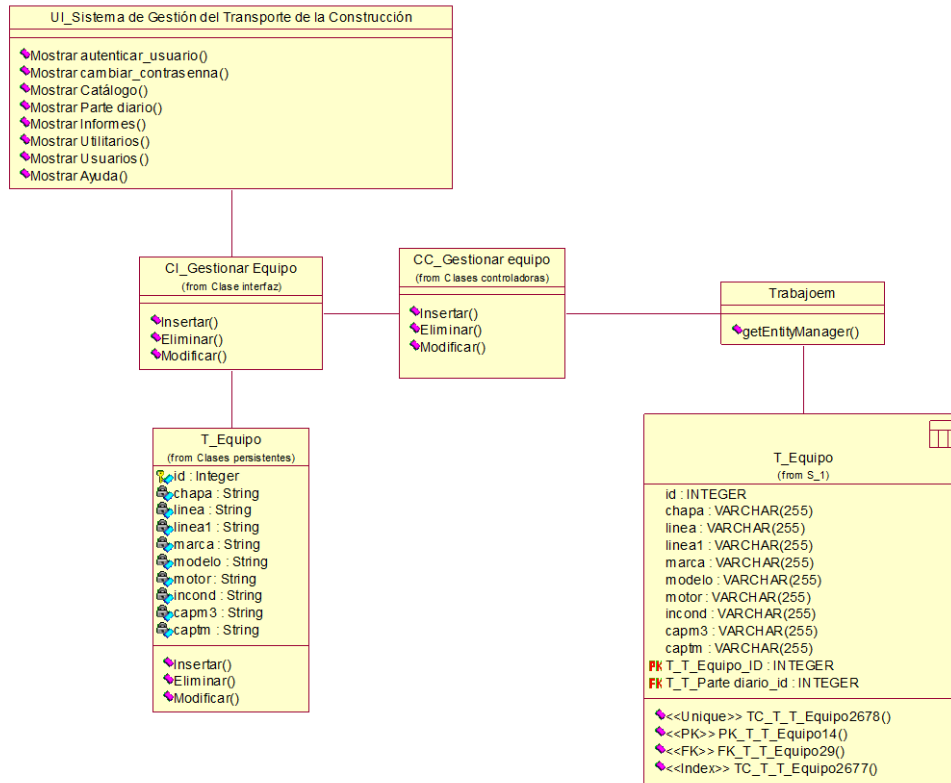


Figura 17 Diagrama de clases del diseño Gestionar equipo

Diagrama de clases del diseño Gestionar municipio

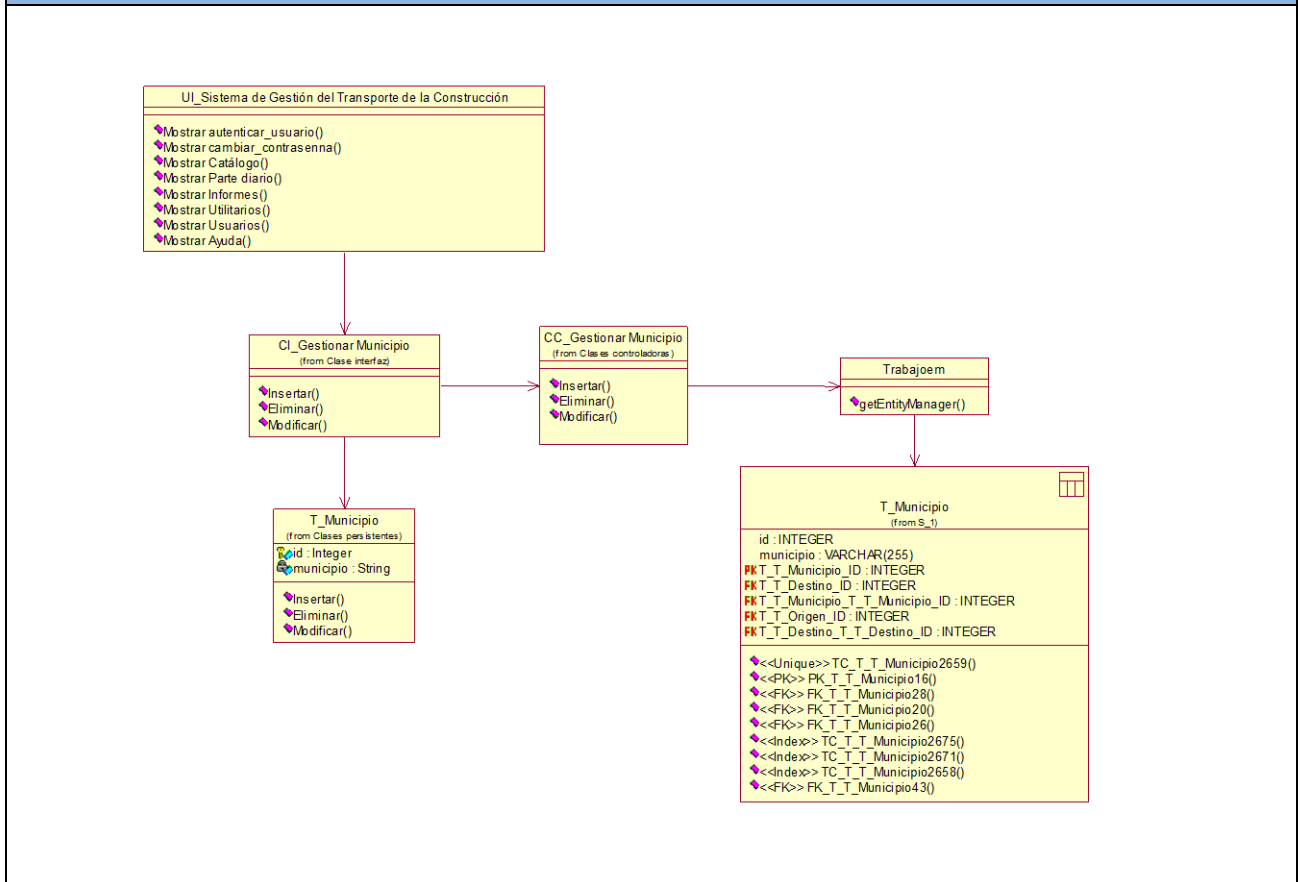


Figura 18 Diagrama de clases del diseño Gestionar municipio

Diagrama de clases del diseño Gestionar origen

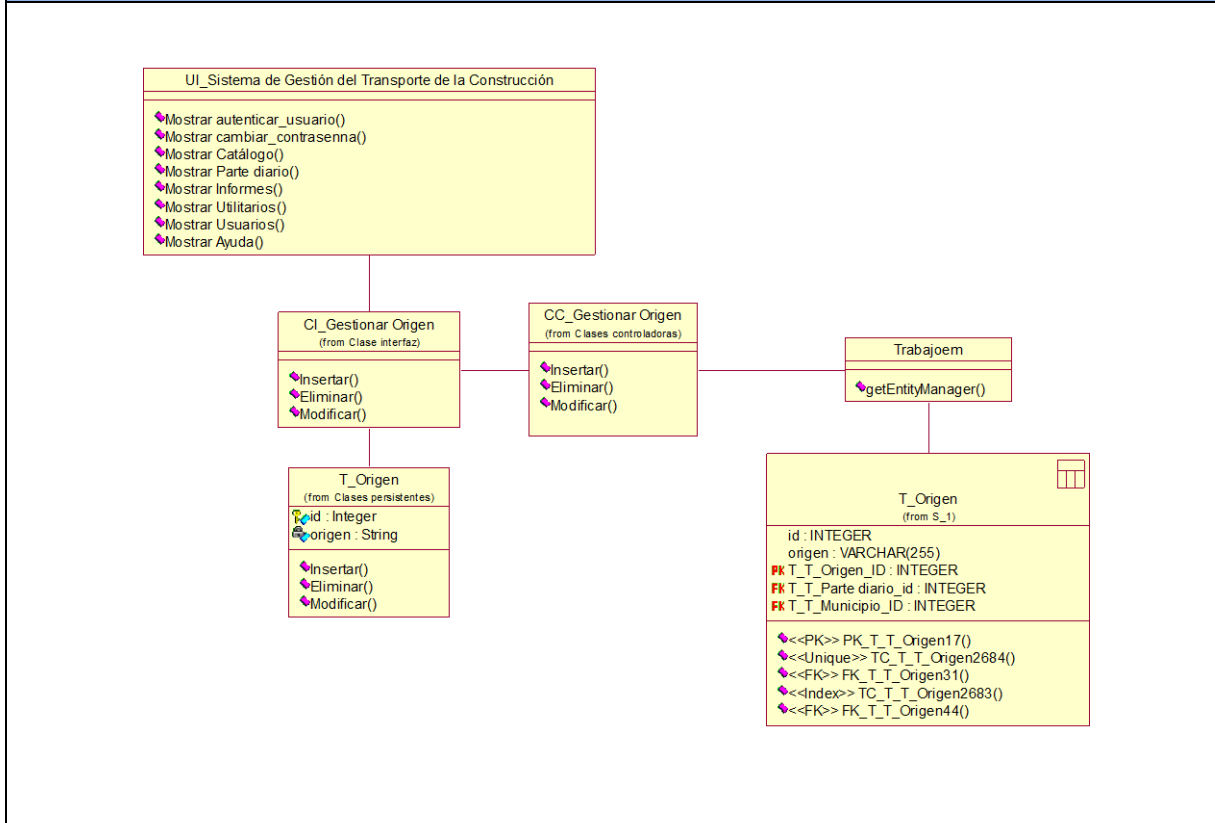


Figura 19 Diagrama de clases del diseño Gestionar origen

Diagrama de clases del diseño Gestionar producto

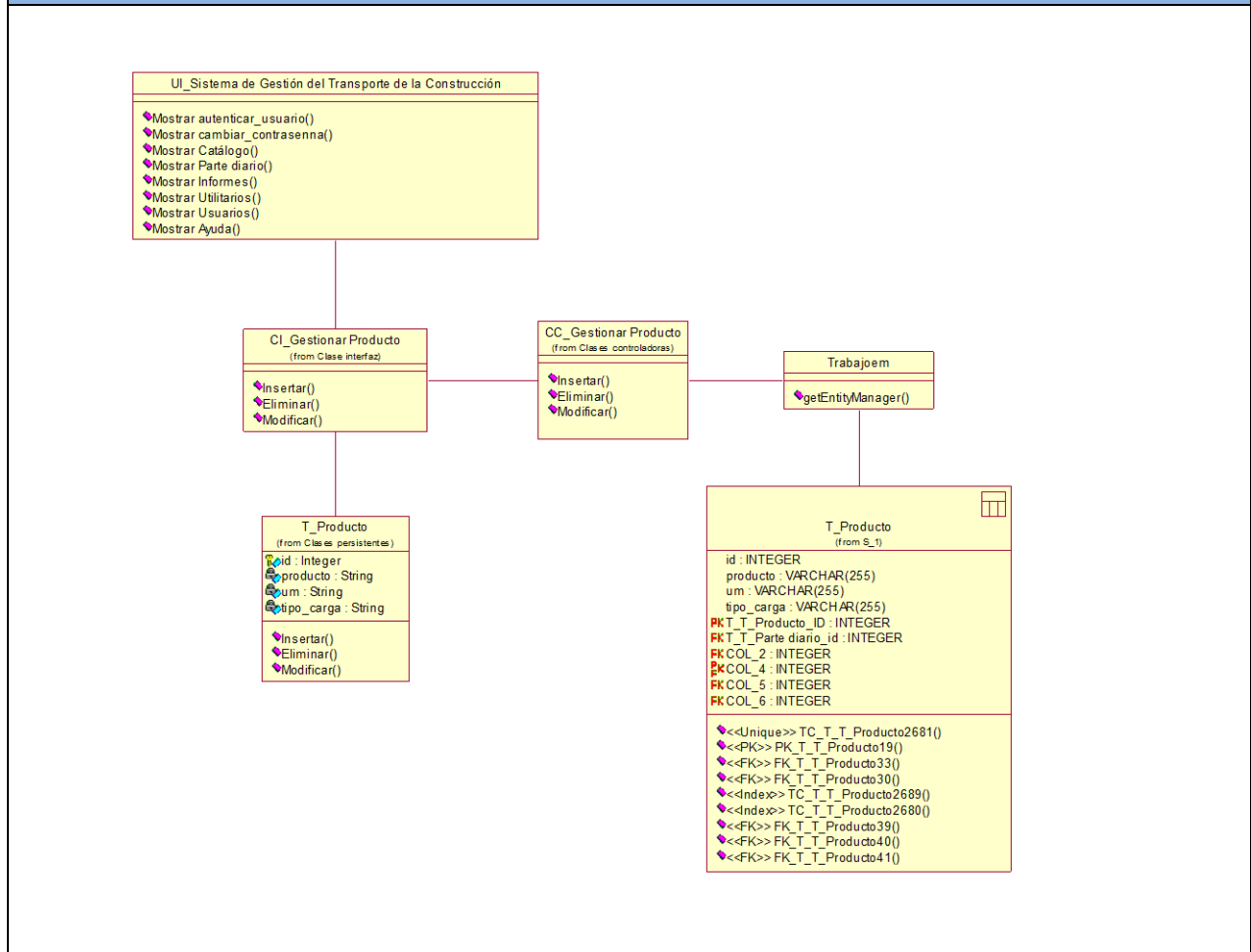


Figura 20 Diagrama de clases del diseño Gestionar producto

Diagrama de clases del diseño Gestionar rol

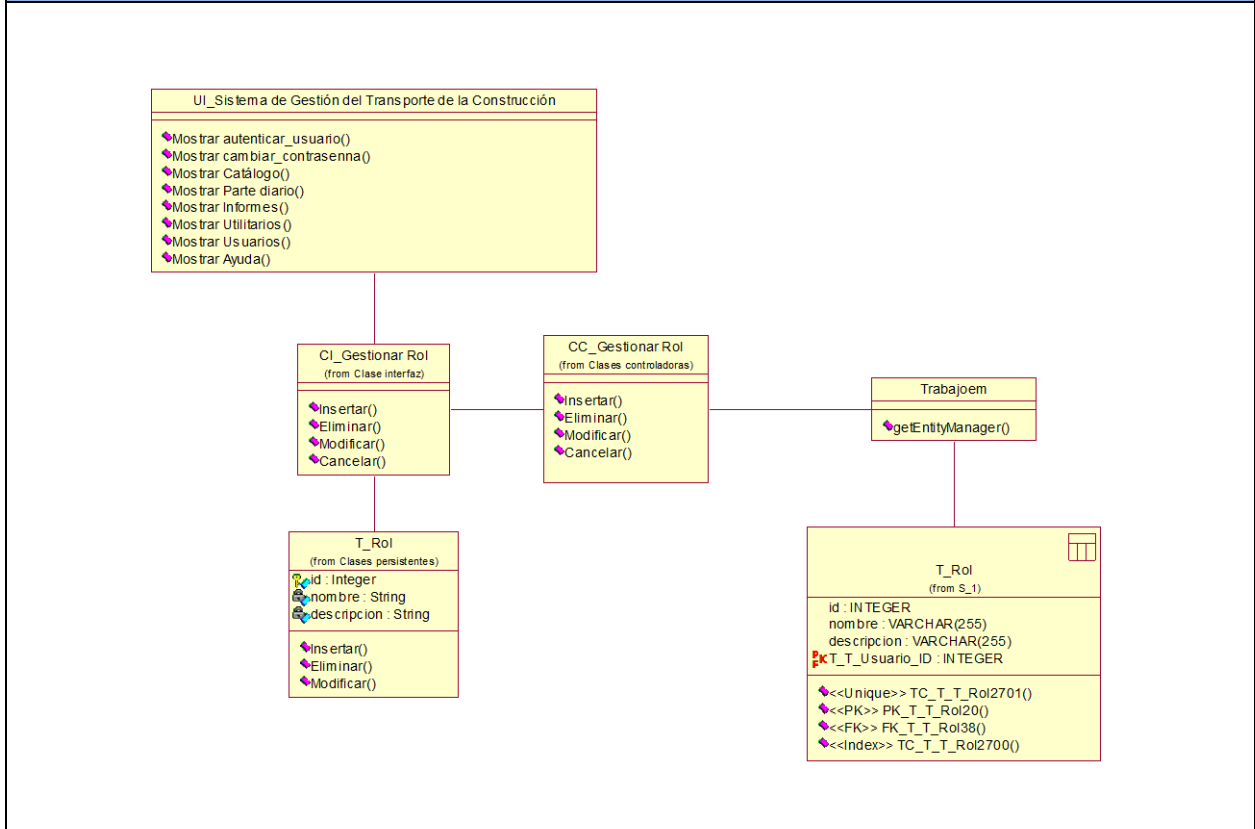


Figura 21 Diagrama de clases del diseño Gestionar rol

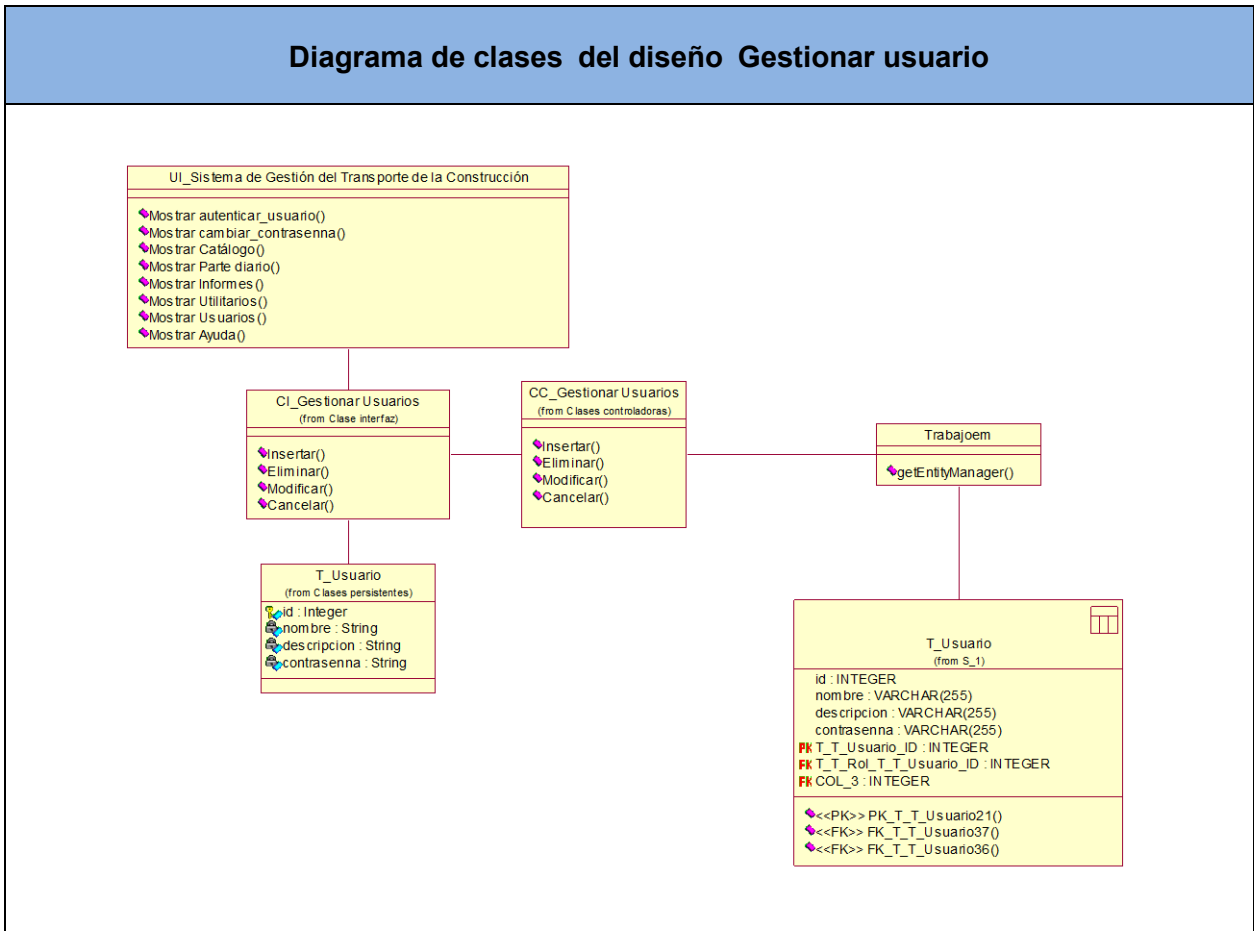


Figura 22 Diagrama de clases del diseño Gestionar usuario

3.2 Diseño de la base de datos

3.2.1 Diagrama de clases persistentes

En el diagrama de clases persistentes aparecen las clases que persisten, las cuales poseen la capacidad de mantener su valor en el espacio y en el tiempo. (Rumbaugh, Booch & Jacobson, 2006). Está compuesto por clases, asociaciones y atributos; interfaces, con sus operaciones y constantes; métodos; información sobre los tipos de atributos, entre otros.

Diagrama de clases persistentes

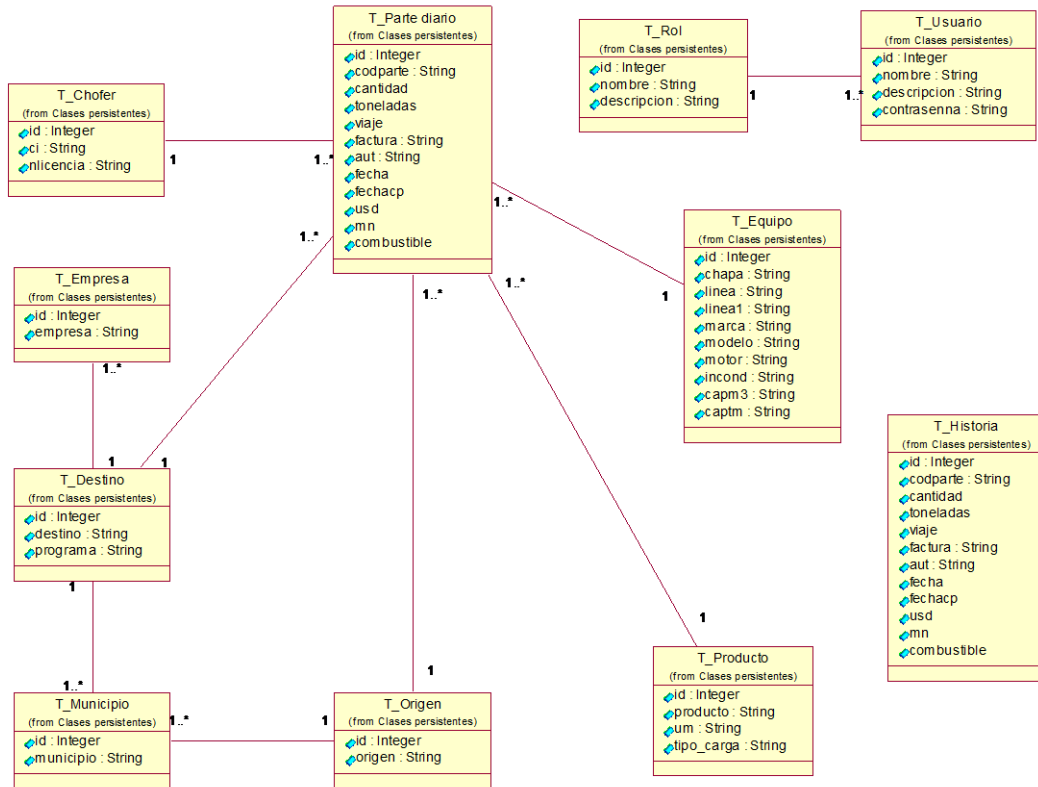


Figura 23 Diagrama de clases persistentes

3.2.2 Modelo de datos

El modelo físico de datos, representa la estructura o descripción física de las tablas de la base de datos, obtenido a partir del modelo lógico de datos (Rumbaugh, Booch & Jacobson, 2006).

encuentran todas las funcionalidades del sistema, el usuario podrá acceder a ellas en dependencia de los permisos que tenga.

3.3.2 Tratamiento de errores

En el sistema propuesto se evitan, minimizan y tratan los posibles errores, con el fin de garantizar la integridad y confiabilidad de la información que se registra y muestra en el sistema en caso de existir algún error se le comunica al usuario a través de un mensaje de alerta en un lenguaje fácil de comprender

3.3.3 Concepción general de la ayuda

El sistema cuenta con una ayuda capaz de explicar al usuario cómo es la manipulación y funcionamiento del software. La ayuda está formada por un menú y dentro de ella existen varios submenús en forma de árbol que contienen todas las funcionalidades del sistema. El usuario podrá navegar por cada uno y consultarlo en cualquier momento que desee ya que se encuentra ubicada en el menú principal de la aplicación.

3.3.4 Seguridad

El sistema mantiene un fuerte mecanismo de seguridad, basado en un nombre de usuario y contraseña para el acceso al mismo. Las personas encargadas de actualizar cualquier información presente en la base de datos del sistema, tendrán nombres de usuario y contraseñas únicas, evitando que esta acción pueda ser realizada por cualquier otro usuario. Además los usuario del sistema tendrán solamente permiso de acceso a las funcionalidades correspondientes a su rol.

3.4 Estándares de codificación

Actualmente se hallan estándares de codificación para la mayoría de los lenguajes existentes. El uso de ellos partiendo de las convenciones definidas permite una mejor comunicación entre los programadores creando las condiciones para la reusabilidad y el mantenimiento de los sistemas. Por lo que se decide que las variables, nombres de funciones, de consultas y objetos del documento son cortos, claros y describen su propósito. Los nombres de las clases se escriben con mayúscula, las variables con minúsculas y las funciones que están compuestas por más de una palabra se escribe primero con minúscula y la primera letra de las demás palabras con mayúscula. Los objetos o tipos de control se nombran según el valor de su contenido. Los inicios y cierre de ámbito se encuentran alineados debajo de la declaración a la

que pertenecen y se evitan si hay sólo una instrucción. Los signos lógicos y de operación se separan por un espacio antes y después de los mismos.

3.5 Modelo de implementación

El modelo de implementación describe cómo los elementos del modelo del diseño, se implementan en términos de componentes, describe cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en los lenguajes de programación utilizados y cómo dependen unos componentes de otros (Jacobson, Booch & Rumbaugh, 2000).

3.5.1 Diagrama de despliegue

El diagrama de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuyen las funcionalidades entre los nodos de cómputo (Jacobson, Booch, & Rumbaugh, 2000). El modelo representa dos nodos: en el lado del servidor se encuentra PostgreSQL como servidor de base de datos y se comunica con el cliente a través del protocolo TCP/IP. Se visualiza la aplicación en el lado del cliente.

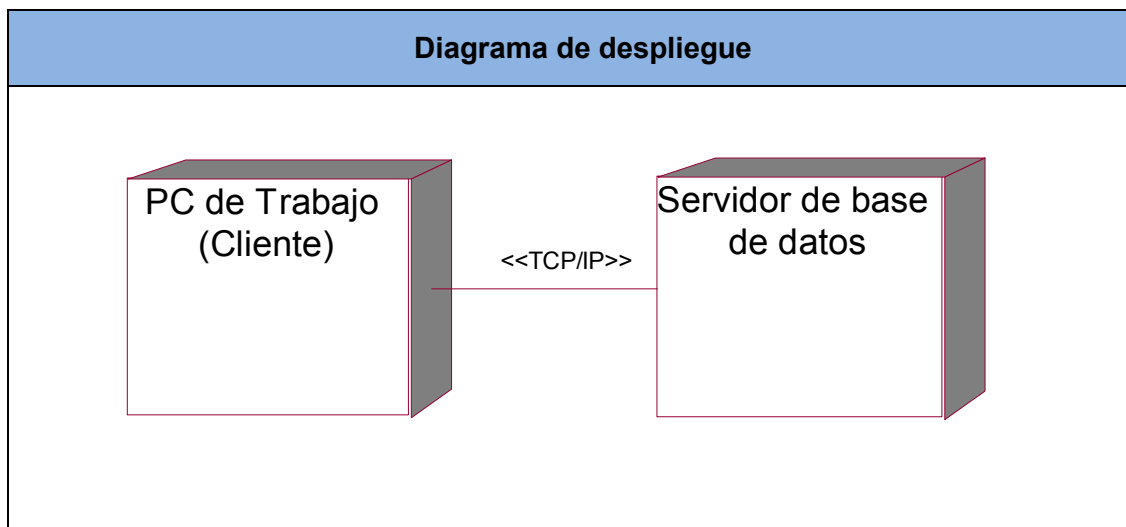


Figura 25 Diagrama de despliegue

3.5.2 Diagrama de componentes

Un diagrama de componentes muestra un conjunto de elementos del modelo, se utiliza para modelar la vista estática de un sistema, muestra la organización y dependencias lógicas entre los componentes del software ya sean bibliotecas, ejecutables o componentes binarios.

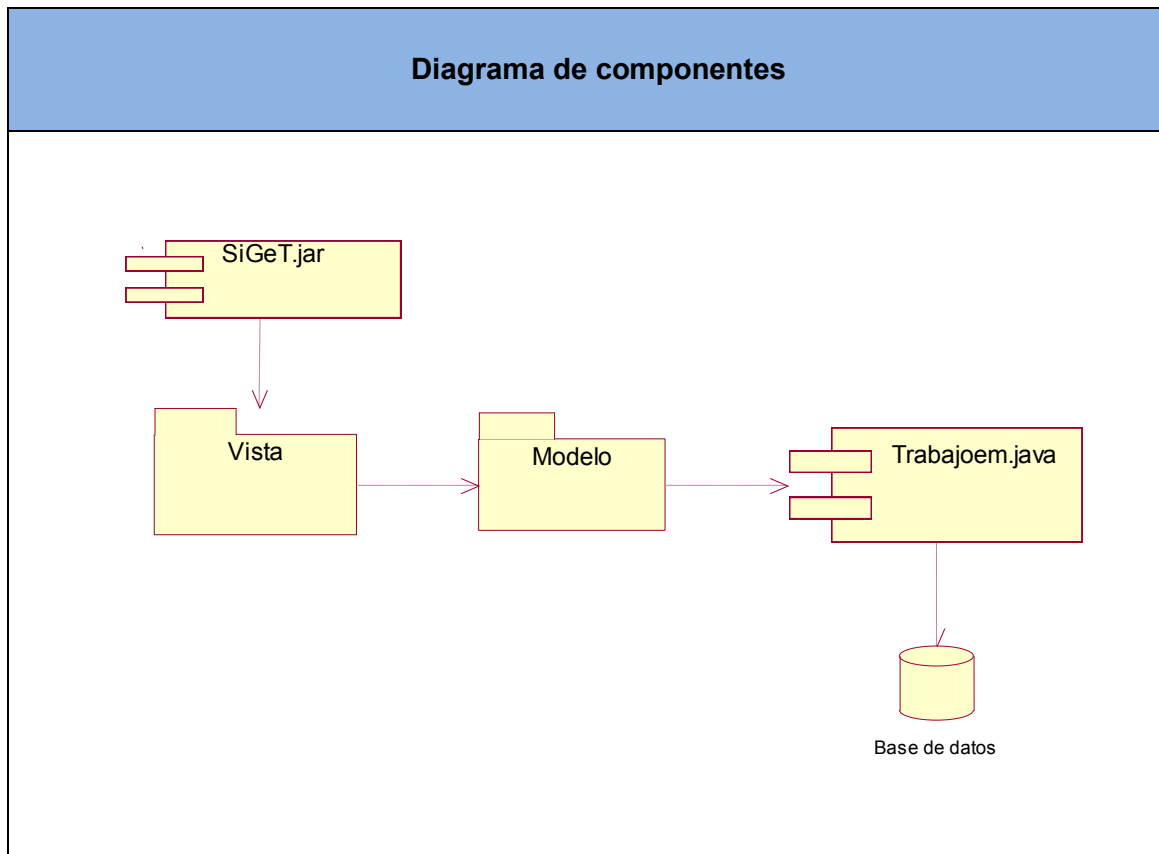


Figura 26 Diagrama de componentes

Componente	Descripción
SiGeT	Este componente representa el ejecutable de la aplicación.
Vista	Dentro de este subsistema se encuentran todas las clases que muestran una interfaz al usuario.
Modelo	Este subsistema representa la lógica del negocio, dentro de él se encuentran las clases entidades con las entidades del negocio y sus atributos.
Trabajoem.java	Este componente representa la clase que facilita el acceso a datos a través de la cual se van a comunicar las clases entidades del modelo y la base de datos. Este componente es usado desde el subsistema Modelo.
Base de Datos	Representa la base de datos de la aplicación.

Tabla 6 Descripción del diagrama de componentes

Conclusiones

Durante el desarrollo de este capítulo se elaboró el diagrama de componente en el cual se graficó de manera clara las relaciones entre los subsistemas, los componentes y la base de datos; el diagrama de despliegue y los modelos lógicos y físicos de la base de datos. Además se describieron los principios de diseño seguidos, específicamente el diseño de la interfaz de usuario, los estándares de codificación, la concepción de la ayuda, el tratamiento de excepciones y la seguridad del sistema.

Conclusiones Generales

Con la realización del presente proyecto se arribó a las siguientes conclusiones:

1. Se determinó mediante el estudio de los fundamentos teóricos y metodológicos la necesidad de creación de una nueva herramienta informática que con el empleo de el lenguaje java para su implementación y PostgreSQL como sistema gestor de bases de datos contribuya a la gestión de la información de la Empresa de Transporte de la Construcción de Sancti Spíritus y se ajuste en mayor grado a los intereses de la institución.
2. Se diseñó una aplicación siguiendo la metodología RUP y apoyada en el lenguaje de modelado UML, permitiendo la descripción de los procesos del negocio, la modelación de los casos de uso correspondientes al sistema, así como el esclarecimiento de las relaciones entre los actores, dando la posibilidad de una adecuada documentación de la solución propuesta.
3. Se implementó una aplicación guiada por los principios del diseño, los estándares de codificación y ajustada a los requerimientos funcionales del sistema, utilizándose el lenguaje de programación Java, como herramienta el IDE Netbeans y como Sistema Gestor de Base de Datos a PostgreSQL por ser tecnologías de código libre y tener las características de integrarse para funcionar en cualquier plataforma.

Recomendaciones

Agregar nuevas funcionalidades al sistema para propiciar su desarrollo y evolución como:

1. El módulo de facturación para así poder tener una constancia del cobro a las empresas que se les prestó el servicio.
2. Incorporar módulos que permitan tener información acerca de los trabajadores no directos a la producción.
3. Adicionar al módulo de solicitud de contrato campos que permitan obtener más información acerca de las empresas que solicitan el servicio de transportación de mercancía para así dar informes más detallados.

Bibliografía

- Ajá, L. (2006). Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones. Obtenido de [www. Monografias.com/trabajo6/tenpe/tenpe](http://www.monografias.com/trabajo6/tenpe/tenpe).
- Bartle, P. (2009). Información para la gestión y gestión de la información. Consultado el 6 de Febrero de 2013, de www.scn.org/mpfc/modules/mon-miss.htm
- Bauer, C., & King, G. (2005). Hibernate in Action. Manning Publications Co.
- Buch, Tomás (1999). Consultado 23 de marzo del 2013. Sistemas tecnológicos; Editorial Aique; Buenos Aires (Argentina);
- Camallea, N., & Abalo, C. (2005). Diccionario de Informática. La Habana: Edición Científico-Técnica.
- Capote Marrero, B. (2006). Consultado el 6 Febrero de 2013, de La gestión de información como herramienta fundamental en el desarrollo de los centros toxicológicos.: www.biblioteca.utp.edu.co/perfildsi.html.
- Carpio Tejera, M. (2010). Software para la gestión de la información relacionada con el proceso productivo de la Fábrica de Cemento Siguaney, Sancti Spíritus. Trabajo de diploma para optar por el título de Ingeniería en Informática, Facultad de Ingeniería, Universidad de Sancti Spíritus "José Martí", Sancti Spíritus, Cuba.
- CMeadors, L., Goodin, B., & Clinton, B. IBATIS in Action.
- Chaín, C. (2007). Gestión de información para la investigación: Desarrollo y consolidación de las funciones del gestor como asesor y consultor para la ciencia y la técnica. Ciencias de la Información.
- Cubagob. Consultado 19 de febrero del 2013, http://www.cubagob.cu/des_eco/mitrans/organism.html
- *Curto, J. (2006). ¿Qué es la gestión de la información? Information Management. Reflexiones sobre las tecnologías de la información. Consultado el 12 de Febrero del 2013, de informationmanagement.wordpress.com*
- Dehesa, Guillermo de la (2008), Comprender la Inmigración, Madrid: Alianza Editorial.
- Eclipse. (2010). Consultado el 11 de febrero del 2013, de <http://www.eclipse.org>
- Free Software Foundation. (2008). Consultado el 11 de Febrero del 2013, de Free software definition.: <http://www.gnu.org/philosophy/free-sw.html>

- González, A. (2005). Modelamiento del negocio. Centro de Estudios de Ingeniería de Sistemas (CEIS).
- Gutiérrez, J. J. (2006). ¿Qué es un framework web? . Recuperado el 15 de mayo de 2011, de <http://jordisan.net/blog/2006/que-es-un-framework/> ¿Qué es un framework?
- Hibernate. (13 de Febrero de 2009). Consultado el 16 de marzo de 2013, de <http://www.hibernate.org>
- Jacobson, I., Booch, G., & Rumbaugh, J. (2000). El proceso unificado de desarrollo. Madrid: Addison Wesley.
- Johnson, A. (2007). PostgreSQL. Recuperado el 10 de Abril de 2013, de Software Libre: http://softwarelibre.net63.net/?page_id=69
- Mas Hernández, J. (2005). Software libre: técnicamente viable, económicamente sostenible y socialmente justo. Consultado el 10 de abril de 2013, de Infonomía red de innovadores: <http://www.infonomia.com>
- Mato García, R. M. (2006). Sistema de Base de Datos. La Habana: Félix Varela.
- Netbeans. (2009). Consultado el 14 de marzo de 2013, de <http://www.netbeans.org>
- Pinto Molina, M. (2001). Gestión de calidad en documentación. Anales de documentación. Consultado el 25 de marzo de 2013, de http://microcampus/web_bib/prod_cientifica/contenido/maestria/trabajos/m-direccion.htm
- Ponjuán Dante, G. (1998). Gestión de información en las organizaciones: principios, conceptos y aplicaciones. Santiago de Chile: Universidad de Chile.
- Popkin Software and System. (2005). Consultado el 25 de marzo de 2013, de Modelado de sistemas con UML: <http://es.tldp.org/tutoriales/doc-modelado-sistemas-UML-multiple-html/c124.html>
- Real Academia Española. (2001). Consultado el 25 de marzo de 2013, de www.raes.es
- Reenskaug, Trygve (1979). Consultado el 23 de marzo del 2013, de <http://www.sceu.frba.utn.edu.ar>
- Rumbaugh, J., Booch, G., & Jacobson, I. (2006). El proceso unificado de desarrollo de software. La Habana: Félix Varela.
- Schildt, H. (2005). Java™: A Beginner's Guide. New York: McGraw-Hill.
- Stallman Richard M. *Software libre para una sociedad libre*, Madrid: Editorial: Traficantes de Sueños, 2004.

- Tanenbaum, Andrew S. (2003). *Redes de computadoras* (4ª edición)
- Truel, A. (2006). Consultado el 4 de abril de 2013, de Introducción a la arquitectura de capas: <http://www ldc.usb.ve/~teruel/ci3715/clases/arqCapas.html>
- Ugaz, M. (s.f.) Consultado el 4 de abril de 2013, de <http://www.usmp.edu.pe/publicaciones/boletin/fia/info36/proyectos.html>

Anexos

Anexo 1: Modelo de objetos



Anexo 2 Descripción de los casos de uso por Paquetes

Anexo 2.1 Paquete Seguridad

CU1 Autenticar usuario	
Actores	Usuario (inicia)
Propósito	Controlar el acceso al software.
Resumen: Este subsistema representa la lógica del negocio, dentro de él se encuentran las clases entidades con las entidades del negocio y sus atributos.	
Referencias	RF-1
Prototipo	Anexo 3

CU2 Cambiar contraseña	
Actores	Usuario (inicia)
Propósito	Mantener la seguridad del sistema y la confiabilidad de los datos al ser la contraseña del usuario únicamente de su conocimiento.
Resumen: El caso de uso comienza cuando el usuario solicita el cambio de su contraseña, introduce los datos pertinentes y de ser correctos el sistema guarda la nueva contraseña, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF-2
Prototipo	Anexo 4

Anexo 2.2 Paquete Administración

CU3 Gestionar usuario	
Actores	Administrador (inicia)
Propósito	Tener controlado el acceso al software a través de usuarios con determinados privilegios.
Resumen: comienza cuando el administrador accede al menú Gestionar usuarios para introducir un nuevo usuario o para modificar o eliminar uno existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF-3,4,5
Prototipo	Anexo 5

Anexo 2.3 Paquete Catálogo

CU4 Gestionar rol	
Actores	Administrador (inicia)
Propósito	Tener controlado el acceso al software a través de usuarios con determinados privilegios.
Resumen: comienza cuando el administrador accede al menú Gestionar rol para introducir un nuevo rol o para modificar o eliminar uno existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF-6,7,8
Prototipo	Anexo 6

CU5 Gestionar municipio	
Actores	Técnico (inicia)
Propósito	Tener digitalizados cada uno de los municipios donde hayan solicitado el servicio y así facilitar el trabajo.
Resumen: comienza cuando el técnico accede al menú Gestionar municipio para introducir un nuevo municipio o para modificar o eliminar uno existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF-9,10,11
Prototipo	Anexo 7

CU6 Gestionar empresa	
Actores	Técnico (inicia)
Propósito	Tener digitalizadas cada una de las empresas donde hayan solicitado el servicio de nuestra entidad y así facilitar el trabajo.
Resumen: comienza cuando el técnico accede al menú Gestionar empresa para introducir una nueva empresa o para modificar o eliminar alguna existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF-12,13,14
Prototipo	Anexo 8

CU7 Gestionar equipo	
Actores	Técnico (inicia)
Propósito	Tener digitalizados cada uno de los equipos que operan los choferes para así controlarlos y hacer el trabajo más fácil.
Resumen: comienza cuando el técnico accede al menú Gestionar equipo para introducir un nuevo equipo o para modificar o eliminar uno existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF-15,16,17
Prototipo	Anexo 9

CU8 Gestionar origen	
Actores	Técnico (inicia)
Propósito	Tener digitalizado el origen para así saber con detalle del lugar de donde salen los materiales que transporta la entidad.
Resumen: comienza cuando el técnico accede al menú Gestionar origen para introducir un nuevo origen o para modificar o eliminar uno existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF-18,19,20
Prototipo	Anexo 10

CU9 Gestionar destino	
Actores	Técnico (inicia)
Propósito	Tener digitalizado el destino para así saber con detalle del lugar hacia donde van los materiales que transporta la entidad.
Resumen: comienza cuando el técnico accede al menú Gestionar destino para introducir un nuevo destino o para modificar o eliminar uno existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF-21,22,23
Prototipo	Anexo 11

CU10 Gestionar producto	
Actores	Técnico (inicia)
Propósito	Tener digitalizado el producto para tenerlos controlados y facilitar el trabajo.
Resumen: comienza cuando el técnico accede al menú Gestionar producto para introducir un nuevo producto o para modificar o eliminar uno existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF-24,25,26
Prototipo	Anexo 12

CU11 Gestionar parte diario	
Actores	Técnico (inicia)
Propósito	Al tener toda esta información digitalizada en el parte diario podemos tener un resumen de todo lo que se va llevando a cabo en la entidad.
Resumen: comienza cuando el técnico accede al menú Gestionar parte diario para introducir un nuevo parte diario o para modificar o eliminar uno existente, luego introduce los datos pertinentes, de ser correctos el sistema ejecuta la acción, en caso contrario emite un mensaje de error. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF-27,28,29
Prototipo	Anexo 13

Anexo 2.4 Paquete Reportes

CU12 Reporte de listado de choferes	
Actores	Usuario (inicia)
Propósito	Ver en un informe cada uno de los choferes que trabajan en la entidad.
Resumen: comienza cuando el usuario accede al menú Informes y da clic en el submenú Reporte de listado de choferes para ver la información existente de no existir ninguna el sistema le dará la información.	
Referencias	RF-30
Prototipo	Anexo 14

CU13 Reporte de parte diario	
Actores	Usuario (inicia)
Propósito	Ver un informe completo sobre la producción de la empresa.
Resumen: comienza cuando el usuario accede al menú Informes y da clic en el submenú Reporte de parte diario para ver la información existente de no existir ninguna el sistema le dará la información.	
Referencias	RF-31
Prototipo	Anexo 15

CU14 Reporte de productos transportados al MICONS	
Actores	Usuario (inicia)
Propósito	Ver un informe sobre los productos que se le han transportado al MICONS.
Resumen: comienza cuando el usuario accede al menú Informes y da clic en el submenú Reporte de productos transportados al MICONS para ver la información existente de no existir ninguna el sistema le dará la información.	
Referencias	RF-32
Prototipo	Anexo 16

CU15 Reporte de productos transportados por obra	
Actores	Usuario (inicia)
Propósito	Ver un informe sobre los productos que se han transportado por cada obra de construcción unida al servicio de la entidad.
Resumen: comienza cuando el usuario accede al menú Informes y da clic en el submenú Reporte de productos transportados por obra para ver la información existente de no existir ninguna el sistema le dará la información.	
Referencias	RF-33
Prototipo	Anexo 17

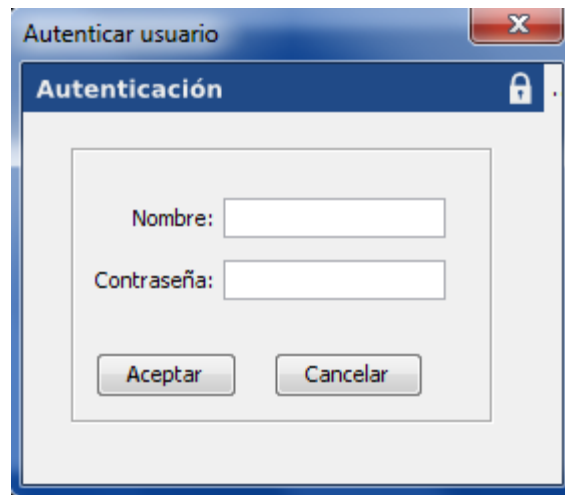
CU16 Reporte de actividades por tipo de carga y programa	
Actores	Usuario (inicia)
Propósito	Ver un informe sobre las actividades que se han realizado por tipo de carga y programa.
Resumen: comienza cuando el usuario accede al menú Informes y da clic en el submenú Reporte de actividades por tipo de carga y programa para ver la información existente de no existir ninguna el sistema le dará la información.	
Referencias	RF-34
Prototipo	Anexo 18

CU17 Reporte por línea de equipo	
Actores	Usuario (inicia)
Propósito	Ver un informe sobre las líneas de equipos utilizadas en la entidad.
Resumen: comienza cuando el usuario accede al menú Informes y da clic en el submenú Reporte por línea de equipo para ver la información existente de no existir ninguna el sistema le dará la información.	
Referencias	RF-35
Prototipo	Anexo 19

CU18 Reporte de resumen por chofer	
Actores	Usuario (inicia)
Propósito	Ver un informe resumido destinado a los choferes.
Resumen: comienza cuando el usuario accede al menú Informes y da clic en el submenú Reporte de resumen por chofer para ver la información existente de no existir ninguna el sistema le dará la información.	
Referencias	RF-37
Prototipo	Anexo 20

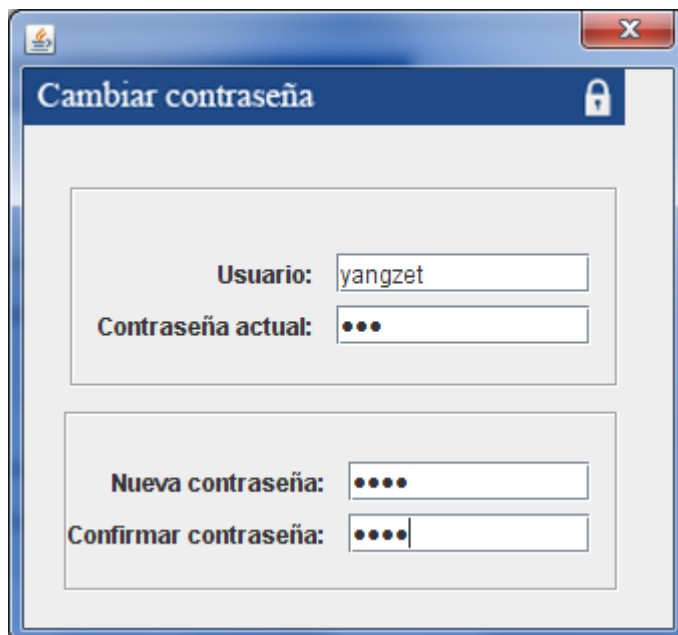
CU19 Reporte de producción por chofer	
Actores	Usuario (inicia)
Propósito	Ver un informe sobre la producción que ha realizado cada chofer en la entidad.
Resumen: comienza cuando el usuario accede al menú Informes y da clic en el submenú Reporte de producción por chofer para ver la información existente de no existir ninguna el sistema le dará la información.	
Referencias	RF-38
Prototipo	Anexo 21

Anexo 3 Prototipo caso de uso Autenticar usuario



The image shows a Windows-style dialog box titled "Autenticar usuario". The title bar includes a close button (X). Below the title bar is a header area with the text "Autenticación" and a lock icon. The main content area contains two text input fields: "Nombre:" followed by an empty text box, and "Contraseña:" followed by an empty text box. At the bottom of the dialog, there are two buttons: "Aceptar" and "Cancelar".

Anexo 4 Prototipo caso de uso Cambiar contraseña



The image shows a Windows-style dialog box titled "Cambiar contraseña". The title bar includes a close button (X). Below the title bar is a header area with the text "Cambiar contraseña" and a lock icon. The main content area is divided into two sections. The top section contains two text input fields: "Usuario:" with the text "yangzet" entered, and "Contraseña actual:" with three dots (•••) entered. The bottom section contains two text input fields: "Nueva contraseña:" with four dots (••••) entered, and "Confirmar contraseña:" with four dots (••••) entered.

Anexo 5 Prototipo caso de uso Gestionar usuarios

The screenshot shows a window titled "Gestionar Usuarios". It contains a table with the following data:

Nombre	Descripción	Rol
admin	Administrador	Administrador

Below the table is a form with the following fields:

- Nombre:
- Descripción:
- Contraseña:
- Repetir contraseña:
- Rol:

At the bottom of the window are several icons: a plus sign (+), a red X, a pencil (edit), a yellow arrow (undo), and two arrows (left and right).

Anexo 6 Prototipo caso de uso Gestionar rol

The screenshot shows a window titled "Roles de Usuario". It contains a table with the following data:

Nombre	Descripción
Administrador	Acceso total al sistema
Invitado	Ver reportes
Técnico	Gestión de datos

Below the table is a form with the following fields:

- Nombre:
- Descripción:

At the bottom of the window are several icons: a plus sign (+), a red X, a pencil (edit), a yellow arrow (undo), and two arrows (left and right).

Anexo 7 Prototipo caso de uso Gestionar municipio

Introducir Municipio

Municipio
Fomento
La Sierpe
Jatibonico
Cabaiguán
Taquasco

Municipio:

Navigation icons: [Back] [Previous] [Add] [Delete] [Edit] [Next] [Forward]

Anexo 8 Prototipo caso de uso Gestionar empresa

Introducir Empresa

Empresa
ENPA
EPASE
EMPROVA
Recursos Hidráulicos

Empresa:

Navigation icons: [Back] [Previous] [Add] [Delete] [Edit] [Next] [Forward]

Anexo 9 Prototipo caso de uso Gestionar equipo

The screenshot shows a software window titled "Introducir datos de Equipo". It contains a table with the following data:

Chapa	Linea	Linea1	Marca	Modelo	Motor	Indcond	Cap M3	Cap TM
SSF235	CP	CP	Pegaso	1086	Pegaso	2.38	0	25.0
SSF564	CP	CP	Pegaso	1087	Pegaso	2.45	1	30.0
SSF234	CP	CP	Pegaso	1086	Pegaso	3.01	2	35.0

Below the table are input fields for the selected row (SSF234):

Chapa: SSF234 Marca: Pegaso Indcond: 3.01
Linea: CP Modelo: 1086 CapM3: 2
Línea 1: CP Motor: Pegaso CapTM: 35

At the bottom, there is a navigation bar with icons for back, previous, add, delete, edit, next, and forward.

Anexo 10 Prototipo caso de uso Gestionar origen

The screenshot shows a software window titled "Introducir datos de Origen". It contains a table with the following data:

Municipio	Origen
Fomento	MICONS
Jatibonico	GECSS
Taguasco	MICONS
La Sierpe	GECSS

Below the table are input fields for the selected row (Fomento):

Origen: Municipio: Fomento

At the bottom, there is a navigation bar with icons for back, previous, add, delete, edit, next, and forward.

Anexo 11 Prototipo caso de uso Gestionar destino

The screenshot shows a window titled "Introducir datos de Destino". It contains a table with the following data:

Destino	Programa	Empresa	Municipio
Viv. Circunv. Sur	MICONS		Fomento
Viv. Circunv. N...	MICONS		Cabaiguán

Below the table are four form fields:

- Destino:
- Programa:
- Municipio:
- Empresa:

At the bottom of the window is a toolbar with the following icons: a left arrow, a double left arrow, a plus sign, a red X, a pencil, a double right arrow, and a right arrow.

Anexo 12 Prototipo caso de uso Gestionar producto

The screenshot shows a window titled "Introducir datos de Producto". It contains a table with the following data:

Producto	UM	Tipocarga
Arena	23	General
Piedra	56	General
Cabilla	60	General
Cemento	34	General

Below the table are three form fields:

- Producto:
- UM:
- Tipo de Carga:

At the bottom of the window is a toolbar with the following icons: a left arrow, a double left arrow, a plus sign, a red X, a pencil, a double right arrow, and a right arrow.

Anexo 13 Prototipo caso de uso Gestionar parte diario

Introducir datos del Parte Diario

Codparte	Origen	Destino	Producto	Chofer	Equipo
1	MICONS	Viv. Circunv. Norte	Arena	Raúl Méndez	SSF234
2	GECSS	Viv. Circunv. Norte	Cemento	Eusebio Rodríguez	SSF234

Codparte: Equipo:

Producto: Chofer:

Destino: Cantidad:

Origen: Toneladas:

Viaje: Fecha:

Factura: FechaCp:

Aut: USD:

Combustible: MN:

Anexo 14 Prototipo caso de uso Reporte de listado de choferes

Listado de Chóferes			Fecha: 29/05/13 11:50 PM
CI	Nombre	No. Licencia	
56061316578	Raúl Méndez	8B34567	
66121345678	Orlando García	5T 65433	
81101813426	Eusebio Rodríguez	9A95889	
78012312345	Pedro García	1A21355	
67120312456	Luis Pérez	9A34567	
65120412678	Rolando González	9A46565	
69120345678	Reinaldo oárez	9A46578	

Anexo 15 Prototipo caso de uso Reporte de parte diario

Parte Diario							Fecha Inicio: 28/05/13 10:35 AM
							Fecha Fin: 28/05/13 10:35 AM
Fecha	Código	Producto	Origen	Destino	Cantidad	Toneladas	Viajes
1/05/13 12:00 AM	1	Arena	MICONS	Viv. Circunv. Norte	3.00	56.00	4
7/05/13 12:00 AM	2	Cemento	GECSS	Viv. Circunv. Norte	2.00	2.00	2
4/05/13 12:00 AM	1	Arena	MICONS	Viv. Circunv. Sur	12.00	32.00	23

Anexo 16 Prototipo caso de uso Reporte de productos transportados al MICONS

Productos transportados al Micons			
			Fecha Inicio: 28/05/13 10:39 AM
			Fecha Fin: 28/05/13 10:39 AM
Producto	Cantidad	Toneladas	Combustible
Arena	12.00	32.00	13.00

Anexo 17 Prototipo caso de uso Reporte de productos transportados por obra

Productos transportados por obra			
			Fecha Inicio: 28/05/13 10:46 AM
			Fecha Fin: 28/05/13 10:46 AM
Destino: Vlv. Circunv. Norte			
Producto	Cantidad	Toneladas	Combustible
Arena	3.00	56.00	40.00
Cemento	2.00	2.00	4.00
Destino: Vlv. Circunv. Sur			
Producto	Cantidad	Toneladas	Combustible
Arena	12.00	32.00	13.00

Anexo 18 Prototipo caso de uso Reporte de actividades por tipo de carga y programa

Actividades(Tipo de Carga/Programa				
			Fecha Inicio:	28/05/13 10:42 AM
			Fecha Fin:	28/05/13 10:42 AM
Tipo de carga	CAPM3	Toneladas	Viajes	Combustible
General	2	56.00	4	40.00
General	2	2.00	2	4.00
General	0	32.00	23	13.00

Anexo 19 Prototipo caso de uso Reporte por línea de equipo

Por línea de Equipo				
			Fecha Inicio:	28/05/13 10:48 AM
			Fecha Fin:	28/05/13 10:48 AM
Línea de equipo	Toneladas	USD	MN	Combustible
CP	56.00	45.00	23.00	40.00
CP	2.00	4.00	2.00	4.00
CP	32.00	12.00	14.00	13.00

Anexo 20 Prototipo caso de uso Reporte de resumen por chofer

Resumen por Chofer						Fecha Inicio: 28/05/13 11:31 AM
						Fecha Fin: 28/05/13 11:31 AM
CI	Nombre	Toneladas	USD	MN	Combustible	
81101813426	Eusebio Rodríguez	2.00	4.00	2.00	4.00	
56061316578	Raúl Méndez	88.00	57.00	37.00	53.00	

Anexo 21 Prototipo caso de uso Reporte de producción por chofer

Producción X Chofer				Fecha Inicio: 28/05/13 11:34 AM
				Fecha Fin: 28/05/13 11:34 AM
Nombre: Raúl Méndez				CI: 56061316578
Código parte	Destino	Toneladas	Combustible	
1	Viv. Circunv. Norte	56.00	40.00	
2	Viv. Circunv. Norte	2.00	4.00	
1	Viv.Circunv. Sur	32.00	13.00	