

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL



Facultad de Ingeniería en Electricidad y Computación

Maestría en Sistemas de Información Gerencial

“AUTOMATIZACIÓN DE LOS PROCESOS DE BODEGA Y GESTIÓN DE
RECURSOS HUMANOS PARA LA DIRECCIÓN REGIONAL DE LOS ESPACIOS
ACUÁTICOS DE ESMERALDAS UTILIZANDO PLATAFORMA WEB”

TESIS DE GRADO

PREVIO A LA OBTENCIÓN DEL TÍTULO DE:

MAGISTER EN SISTEMAS DE INFORMACIÓN GERENCIAL

DALTON RICHARD NÚÑEZ CARRIÓN

GUAYAQUIL – ECUADOR

AÑO: 2016

AGRADECIMIENTO

A mi familia, fuente inagotable de amor y apoyo permanente e incondicional en mis años de vida y de carrera profesional militar dentro de la Armada del Ecuador, en especial a mi esposa por su imperecedera comprensión y apoyo sacrificando el tiempo de familia juntos, para poder culminar con éxito mis objetivos propuestos.

A la Universidad Politécnica del Litoral, como cuna del conocimiento que, día a día forma profesionales acordes a las exigencias y expectativas de la tecnología actual como reto fundamental para contribuir al desarrollo y engrandecimiento de nuestra Patria.

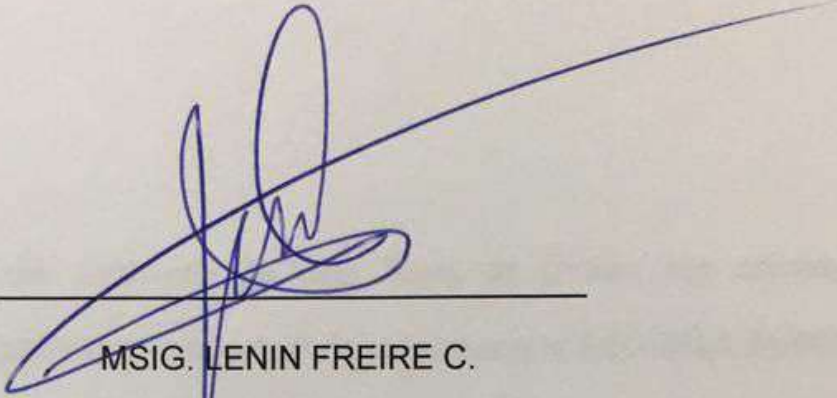
A mis formadores y profesores, que han sido la pieza clave en la elaboración de éste trabajo, ya que sin sus valiosos conocimientos que me han impartido no hubiera sido posible la ejecución del mismo.

DEDICATORIA

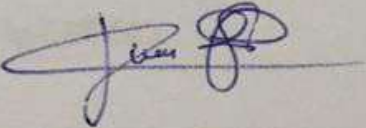
A Dios, por sus bendiciones en mi diario caminar durante mi vida familiar, lo que constituye tranquilidad y fortaleza para seguir adelante persiguiendo mis sueños trazados.

A mi amada esposa, por estar a mi lado apoyándome incondicionalmente constituyéndose en mi mejor amiga, y la mejor esposa del universo, enseñándome que para que las cosas salgan bien, se las debe hacer con infinito amor sabiendo sobrellevar con inteligencia y calma las diferentes dificultades y obstáculos que se presentan.

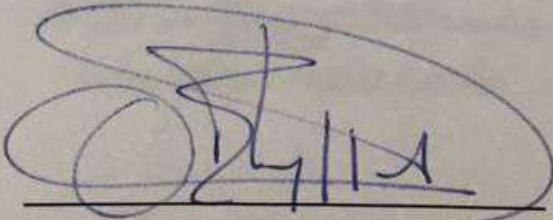
TRIBUNAL DE SUSTENTACIÓN



MSIG. LENIN FREIRE C.
DIRECTOR MSIG



MSIG. JUAN CARLOS GARCÍA P.
DIRECTOR DEL PROYECTO DE GRADUACIÓN




MSIG. OMAR MALDONADO D.
MIEMBRO DEL TRIBUNAL

DECLARACIÓN EXPRESA

"La responsabilidad del contenido de esta Tesis de Grado, me corresponde exclusivamente; y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL".

(Reglamento de Graduación de la ESPOL)



Dalton Richard Núñez Carrión

RESUMEN

Los sistemas informáticos en el sector público principalmente han servido para optimizar importantes procesos, lo que contribuye al mejoramiento de la calidad del servicio a los usuarios finales, reduciendo considerablemente el tiempo de espera para atención, debido a la oportunidad de obtener datos procesados en tiempo real optimizando recursos humanos, materiales, tiempo y dinero.

Para el desarrollo de los diferentes aplicativos que permitan este fin, se busca las mejores herramientas informáticas disponibles, tomando en cuenta las tendencias actuales, las mismas que deberán garantizar una escalabilidad y sostenibilidad durante su ciclo de vida proyectado. En la actualidad, las aplicaciones sobre plataforma web ha ido ganado terreno debido a su rápido y fácil acceso desde cualquier parte del mundo facilitando así, tanto la interactividad con el usuario final, como las tareas de mantenimiento y soporte por parte de los desarrolladores del mismo. Por cuanto, se ha optado por automatizar los procesos de la Dirección Regional de los Espacios Acuáticos de Esmeraldas con un aplicativo basado en software OpenSource de fácil acceso vía web desde los diferentes centros poblados de la provincia de Esmeraldas.

ÍNDICE GENERAL

AGRADECIMIENTO.....	ii
DEDICATORIA.....	iii
TRIBUNAL DE SUSTENTACIÓN.....	iv
DECLARACIÓN EXPRESA.....	v
RESUMEN.....	vi
ÍNDICE GENERAL.....	xii
ABREVIATURAS.....	xii
ÍNDICE DE FIGURAS.....	xiii
ÍNDICE DE TABLAS.....	xiv
CAPÍTULO 1: GENERALIDADES.....	1
1.1 INTRODUCCIÓN.....	1
1.2 ANTECEDENTES.....	2
1.3 DESCRIPCIÓN DEL PROBLEMA.....	5
1.4 OBJETIVOS.....	6
1.4.1 OBJETIVO GENERAL.....	6
1.4.2 OBJETIVOS ESPECÍFICOS.....	6
1.5 JUSTIFICACIÓN.....	6
1.6 ALCANCE.....	7
CAPÍTULO 2: MARCO TEÓRICO.....	8
2.1 SERVIDOR DE APLICACIONES.....	8
2.1.1 SERVIDORES APACHE CON PHP Y MYSQL.....	9
2.1.2 OTROS SERVIDORES.....	9

2.2 LENGUAJE DE PROGRAMACIÓN.....	10
2.2.1 TIPOS DE LENGUAJES DE PROGRAMACIÓN.....	10
2.2.2 LENGUAJES DE PROGRAMACIÓN MÁS UTILIZADOS.....	12
2.3 MÉTRICAS DEL SOFTWARE.....	15
2.3.1 FACTORES DE CALIDAD DEL SOFTWARE.....	15
2.3.2 FACTORES DE CALIDAD MCCALL.....	16
2.4 MARCO DEL DESARROLLO WEB.....	19
2.4.1 OBJETIVOS DEL FRAMEWORK.....	19
2.4.2 ARQUITECTURA.....	21
2.4.3 LISTADO DE PRINCIPALES FRAMEWORKS PHP.....	23
2.5 METODOLOGÍA RUP (RATIONAL UNIFIED PROCESS).....	23
2.5.1 CARACTERÍSTICAS.....	26
2.5.2 FASES DE LA METODOLOGÍA.....	27
2.5.3 CICLO DE VIDA.....	34
2.6 INVENTARIOS.....	36
CAPITULO 3: ANÁLISIS PARA LA IMPLEMENTACIÓN DEL SISTEMA INFORMÁTICO.....	39
3.1 ELECCIÓN DE HERRAMIENTAS DE DESARROLLO Y GESTOR DE BASE DE DATOS.....	39
3.1.1 SERVIDOR DE APLICACIONES WEB APACHE.....	41
3.1.2 PHP.....	42
3.1.3 BASE DE DATOS MYSQL.....	43
3.2 ANÁLISIS DE MÓDULOS DEL SISTEMA.....	44

3.2.1 MÓDULO DE GESTIÓN RECURSOS HUMANOS.....	45
3.2.2 MÓDULO DE GESTIÓN DE BODEGA.....	51
CAPÍTULO 4: DESARROLLO DEL APLICATIVO.....	55
4.1 PLANIFICACIÓN EL SISTEMA.....	55
4.1.1 PROPÓSITO.....	56
4.1.2 VISTA GENERAL DEL SISTEMA.....	56
4.1.3 SUPOSICIONES Y RESTRICCIONES.....	57
4.1.4 ENTREGABLES DEL PROYECTO.....	57
4.1.5 ORGANIZACIÓN DEL PROYECTO.....	61
4.1.6 PLAN DEL PROYECTO.....	63
4.2 ANÁLISIS Y ELABORACIÓN DEL SISTEMA.....	68
4.2.1 DEFINICIÓN DEL SISTEMA.....	69
4.2.2 REQUISITOS BÁSICOS DEL SISTEMA.....	70
4.2.3 ANÁLISIS Y CASOS DE USO.....	73
4.3 DISEÑO LÓGICO DEL SISTEMA.....	81
4.3.1 ARQUITECTURA.....	86
4.3.2 MODELO FÍSICO DE DATOS.....	88
4.4 DESARROLLO DEL SISTEMA.....	93
4.4.1 PLANIFICACIÓN DE LAS ACTIVIDADES DE DESARROLLO.....	93
4.4.2 DESARROLLO Y CODIFICACIÓN.....	94
CAPÍTULO 5: IMPLEMENTACIÓN Y PRUEBAS DEL SISTEMA.....	99
5.1 IMPLEMENTACIÓN.....	99
5.1.1 DIAGRAMA DE SECUENCIAS.....	99

5.1.2 DIAGRAMA DE ACTIVIDADES.....	102
5.2 PRUEBLAS.....	103
5.2.1 CASO DE PRUEBA: SEGURIDAD DE LA APLICACIÓN.....	103
5.2.2 CASO DE PRUEBA: REGISTRO EGRESOS DE MATERIALES.....	105
5.2.3 CASO DE PRUEBA: REGISTRO INGRESO DE MATERIALES.....	107
5.2.4 CASO DE PRUEBA: REPORTE DE INVENTARIO DE BODEGA.....	109
5.2.5 CASO DE PRUEBA: REPORTE DE PERMISOS Y LICENCIAS.....	110
CAPÍTULO 6: ANÁLISIS DE RESULTADOS.....	113
6.1.1 REPORTE DE BODEGA.....	113
6.1.2 ASIGNACIÓN DE LICENCIAS Y PERMISOS.....	115
CONCLUSIONES Y RECOMENDACIONES.....	117
BIBLIOGRAFÍA.....	120

ABREVIATURAS Y SIMBOLOGÍA

DIGREH:	Dirección General de Recursos Humanos de la Armada.
DIGMAT:	Dirección General del Material de la Armada.
DIRGES:	Dirección Regional de los Espacios Acuáticos de Esmeraldas.
DIRNEA:	Dirección Nacional de los Espacios Acuáticos de Esmeraldas.
HTTP:	Hypertext Transfer Protocol, es un protocolo de comunicación para transacciones en la web.
INSGAR:	Inspectoría General de la Armada.
MIDDLEWARE:	Software de computadora que conecta componentes de software o aplicaciones para que puedan intercambiar datos entre éstas.
PHP:	Hypertext Preprocessor, lenguaje de programación de código abierto.

ÍNDICE DE FIGURAS

FIGURA 2.1 PRINCIPALES LENGUAJES DE PROGRAMACIÓN.....	14
FIGURA 3.1 MÓDULOS DEL SISTEMA WEB DIRGES.....	44
FIGURA 3.2 MÓDULO DE GESTIÓN DE RECURSOS HUMANOS.....	45
FIGURA 3.3 MÓDULO ESTRUCTURA ORGÁNICA ORGANIZACIONAL.....	46
FIGURA 3.4 CATEGORÍAS PARA INFORMAC. PERSONAL EMPLEADOS....	47
FIGURA 3.5 CATEGORÍAS PARA INFORM. RELACIONADA A LICENCIAS...	50
FIGURA 4.1 FASES DEL PROYECTO.....	65
FIGURA 4.2 REQUISITOS TECNOLÓGICOS DEL SISTEMA.....	72
FIGURA 4.3 CONTROL DE ACCESO AL SISTEMA.....	73
FIGURA 4.4 CASO DE USO: ADMINISTRACIÓN DE USUARIOS.....	74
FIGURA 4.5 CASO DE USO: GESTIÓN DE LICENCIAS Y PERMISOS.....	76
FIGURA 4.6 CASO DE USO: ADMINISTRACIÓN DE CATÁLOGO.....	77
FIGURA 4.7 CASO DE USO: ENTRADA/SALIDA ARTÍCULOS DE BODEGA..	79
FIGURA 4.8 CASO DE USO: CUENTAS POR PAGAR.....	80
FIGURA 4.9 DISEÑO LÓGICO DEL SISTEMA.....	82
FIGURA 4.10 ARQUITECTURA CLIENTE-SERVIDOR.....	86
FIGURA 4.11 TABLAS CON DATOS DE WEBSITE.....	88
FIGURA 4.12 TABLAS CON DATOS DE MÓDULOS.....	89
FIGURA 4.13 TABLAS CON DATOS DE CLIENTES.....	90
FIGURA 4.14 TABLAS CON DATOS DE PRODUCTOS.....	91
FIGURA 4.15 TABLAS CON DATOS DE PEDIDOS.....	95

FIGURA 4.16 TABLAS CON DATOS DE CONFIGURACIÓN Y CONTACTOS..	92
FIGURA 4.17 ESQUEMA DEL SITIO WEB.....	95
FIGURA 4.18 TRANSMISIÓN DE DATOS ENTRE CAPAS.....	96
FIGURA 4.19 CONFIGURACIÓN DE CONEXIÓN AL ORIGEN DE DATOS.....	97
FIGURA 5.1 DIAGRAMA DE SECUENCIA: ACCESO AL SISTEMA.....	100
FIGURA 5.2 DIAGRAMA DE SECUENCIA: COMPRA A PROVEEDORES.....	100
FIGURA 5.3 DIAGRAMA DE SECUENCIA: ENTREGA DE PRODUCTOS.....	101
FIGURA 5.4 DIAGRAMA DE SECUENCIA: GESTIÓN Y APROBACIÓN DE LICENCIAS.....	101
FIGURA 5.5 DIAGRAMA DE ACTIVIDADES: ACCESO AL SISTEMA.....	102
FIGURA 5.6 DIAGRAMA DE ACTIVIDADES: PEDIDO DE PRODUCTOS.....	102
FIGURA 5.7 DIAGRAMA DE ACTIVIDADES: ASIGNACIÓN DE LICENCIAS...	103
FIGURA 6.1 ANÁLISIS DE IMPACTO MÓDULO BODEGA.....	114
FIGURA 6.2 ANÁLISIS DE IMPACTO MÓDULO LICENCIAS Y PERMISOS...	116

ÍNDICE DE TABLAS

TABLA 1 FACTORES DE CALIDAD MCCALL.....	16
TABLA 2 RELACIÓN DE FACTORES DE CALIDAD.....	17
TABLA 3 PRINCIPALES FRAMEWORKS PHP.....	22
TABLA 4 PERFILES DE USUARIO DEL SISTEMA.....	62
TABLA 5 PLANIFICACIÓN POR FASES.....	63
TABLA 6 CALENDARIO DEL PROYECTO.....	66

CAPÍTULO 1

GENERALIDADES

1.1 INTRODUCCIÓN

Los sistemas informáticos han sido creados con la finalidad de resolver problemas desde los más comunes hasta los más complejos, haciendo las actividades del ser humano más rápidas y eficientes, desde el nacimiento de la tecnología, los procesos administrativos en las organizaciones han sufrido cambios trascendentales, desde el uso del papel pasando por la máquina de escribir hasta el uso de ordenadores y base de datos alojadas en cualquier servidor alrededor del mundo, facilita el acceso de la información, trayendo con esto problemas en cuanto a la seguridad de transmisión de la información así como su confidencialidad, por lo que hoy en día el tema de la seguridad informática cada día se han vuelto más necesaria a fin de proteger uno de los principales activos de cualquier organización, que es la información.

Las plataformas informáticas basadas en la web, así como su desarrollo en aplicaciones de código abierto abren nuevas posibilidades de implementación y desarrollo de software asequible para cualquier pequeña o mediana empresa con recursos limitados, que pretendan mejorar sus procesos, a fin de mantener mayor rentabilidad en menos tiempo siendo más eficientes y eficaces, soportados por una comunidad en crecimiento de desarrolladores de software libre sumado a las políticas internas sobre el limitado uso de software propietario hace cada vez más tentativo su desarrollo e inversión.

1.2 ANTECEDENTES

Las instituciones públicas del estado ecuatoriano, a fin de agilizar sus procesos optimizando recursos, cuenta con diversas aplicaciones informáticas para tal efecto. Es así que la Dirección Regional de Espacios Acuáticos de Esmeraldas (DIRGES), como una institución que presta atención al usuario civil requiere de mencionadas herramientas.

Es política de estado utilizar plataformas OpenSource a fin de abaratar costos y evitar la adquisición de licencias, utilizando sistemas con plataforma web, ya que día a día el acceso al internet está al alcance de los centros poblados denominados como caletas para las embarcaciones del sector pesquero artesanal principalmente.

Las Direcciones Regionales son órganos de ejecución y control subordinados a la Dirección Nacional de los Espacios Acuáticos (DIRNEA), las mismas que ejercen la Autoridad Marítima en el área de su jurisdicción. La DIRGES, al igual que las otras Direcciones Regionales, posee la responsabilidad de velar por la conservación de la vida y el medio ambiente en todas las actividades marítimas y fluviales que se realicen dentro de su jurisdicción.

Entre sus funciones están: Aplicar el marco legal vigente, las políticas y disposiciones de la Dirección Nacional de los Espacios Acuáticos para el transporte por agua y la actividad portuaria, ejercer el control y las funciones de Policía Marítima Fluvial, ejecutar las tareas contempladas en los planes de defensa nacional.

A fin de lograr un control efectivo en todo el perfil costanero dentro de la jurisdicción de Esmeraldas, la DIRGES cuenta con retenes navales asentados en los diferentes centros poblacionales destinados como caletas para las diferentes asociaciones pesqueras artesanales existentes en la provincia. Mencionados retenes dependen tanto de recursos materiales como humanos de la DIRGES.

Las adquisiciones de material y la gestión del talento humano lo gestiona DIRGES ante los organismos superiores como la Dirección de Recursos Humanos de la Armada del Ecuador (DIGREH), la Dirección del Material de

la Armada del Ecuador (DIGMAT). Por tratarse de una institución pública, cuando la situación lo amerita, se realiza las adquisiciones por Subasta Inversa Electrónica o compras por catálogo acorde a las regulaciones del Instituto Nacional de Compras Públicas (INCOP). El ente regulador dentro de la Armada del Ecuador es la Inspectoría General de la Armada (INSGAR).

Los procesos de gestión de bodega y de talento humano en la DIRGES se han venido procesando manualmente sin ningún aplicativo que automatice y optimice recursos, tiempo y dinero, por lo que se requiere frecuentemente actualizar la información generada a diario tanto por los ingresos y egresos de insumos y materiales de la bodega de DIRGES, como también de los permisos y de la situación del talento humano en todos los retenes navales acantonados en la provincia de Esmeraldas.

Para realizar cualquier pedido tanto de material como de personal, el personal solicitante debe obligadamente movilizarse en reiteradas ocasiones hasta la ciudad de Esmeraldas a las instalaciones de DIRGES, lo cual disminuye la disponibilidad del personal operativo en los retenes navales.

La gestión manual del Departamento de Personal no ha permitido contar con información actualizada disponible en el momento por lo que actualmente se llenan fichas personales con la información de su grupo familiar como esposa(o), hijos, cargas familiares, cumpleaños, día del niño, día de la madre, agasajos navideños; dicha información es necesaria para el

departamento de bienestar de la DIRGES por cuanto se realiza periódicamente actividades de integración para el personal naval perteneciente a la DIRGES y todo su grupo familiar. Al no contar oportunamente con esta información de forma automática, muchas veces no se ha podido involucrar a todo el personal en mencionadas actividades.

1.3 DESCRIPCIÓN DEL PROBLEMA

La Dirección Regional de Espacios Acuáticos de Esmeraldas (DIRGES), tiene la necesidad de aprovechar el recurso humano y material optimizando el tiempo que el personal ocupa para realizar trámites administrativos por requerimientos de material y del personal tomando en cuenta que la DIRGES posee dos capitanías de puerto y siete retenes navales a lo largo del perfil costanero de la provincia de Esmeraldas, y que el tiempo para movilizarse desde mencionados puntos hasta la DIRGES en vehículo son aproximadamente de dos a dos horas y media tanto hacia el norte como hacia el sur de la provincia.

El personal que hace éstos trámites, demora hasta 5 horas en realizar la gestión de suministros y bodega. Se requiere implementar un sistema web para la gestión de materiales en la bodega de insumos y para la gestión del talento humano en cuanto a cambios de personal y permisos, a fin de poder optimizar recursos y tiempo.

1.4 OBJETIVOS

1.4.1 Objetivo General

Automatizar los procesos de ingresos y egresos de materiales en la bodega de la DIRGES y la gestión del talento humano en cuanto a cambios de personal y permisos mediante un sistema web.

1.4.2 Objetivos Específicos

- ✓ Crear una base de datos del personal que labora en las capitanías de puerto y retenes navales de Esmeraldas.
- ✓ Crear una base de datos de insumos y materiales para las capitanías de puerto y retenes navales de Esmeraldas.
- ✓ Implementar un módulo de Gestión de Recursos Humanos vía Web para las capitanías de puerto y retenes navales de Esmeraldas.
- ✓ Implementar un módulo de Control de Inventario para llevar un registro de los ingresos y egresos de bodega, generado por el sistema web y debidamente legalizado.

1.5 JUSTIFICACIÓN

Las instituciones del estado ecuatoriano, de acuerdo al decreto 1014 firmado el 10 de abril del 2008, tienen la disposición de implementar herramientas informáticas que automaticen los procesos administrativos a fin de optimizar recursos.

Mencionadas aplicaciones deberán utilizar herramientas de desarrollo gratuitas basadas en OpenSource o de código abierto que permita la reutilización de mencionadas aplicaciones en los campos que se requiera, con la finalidad de evitar la adquisición de licencias costosas y la dependencia de empresas de desarrollo para actualizaciones de software conforme el desarrollo y cambios de procesos para lo que fueron diseñados.

1.6 ALCANCE

El presente proyecto será implementado en la DIRGES, y será en beneficio de todo el personal militar naval asentado en la provincia de Esmeraldas, siendo posible la reutilización del software en otras provincias que cuenten con repartos navales al servicio de la colectividad.

Se implementará un módulo web para la Gestión del Talento Humano así como un Control de Insumos en los retenes navales subordinados a la DIRGES. Mencionado aplicativo será desarrollado en PHP con una base de datos MySQL, y los usuarios podrán acceder desde cualquier computador que tenga un explorador web con acceso a internet. Se utilizará código abierto a fin de poder implementar a futuro módulos que permitan automatizar procesos de otras áreas administrativas y operativas de la organización.

CAPÍTULO 2

MARCO TEÓRICO

2.1 SERVIDOR DE APLICACIONES

Es un servidor que forma parte de una red de computadoras y cuya finalidad es la ejecución de determinadas aplicaciones que proveen servicios a los computadores clientes centralizando la información y las aplicaciones de acuerdo a la jerarquización e importancia de las mismas para una determinada empresa.

Dentro de las ventajas de poseer un servidor de aplicaciones tenemos los servicios de middleware que es Software de conectividad que consiste en un conjunto de servicios que permiten interactuar a múltiples procesos que se ejecutan en distintas máquinas a través de una red [1], permitiendo administrar tareas de mantenimiento y parámetros de seguridad para el acceso a datos desde las estaciones cliente.

2.1.1 Servidores Apache con PHP y MySQL

Un servidor Apache es un servidor web http de código abierto, que sirve de plataforma y referencia para la implementación de servidores web.

Al utilizar código abierto, también utiliza un lenguaje de programación de código abierto que permita el diseño y evaluación de sitios web dinámicos a la interacción servidor-cliente, como lo es PHP [2]. Este lenguaje de programación permite el acceso a datos MySQL solventando peticiones por parte del cliente.

2.1.2 Otros Servidores

Los servicios de IIS fueron diseñados para solventar la necesidad de implementación de servidores de aplicaciones sobre plataformas operativas Windows de Microsoft, permitiendo publicar sitios web locales o remotas facilitando la administración y mantenimiento de forma segura. [3]

Estos servidores web incluyen módulos específicos para procesar determinadas funciones, así tenemos los servicios de Active Server Pages (ASP) y ASP.net. Actualmente se pueden implementar módulos de otros fabricantes como PERL o PHP.

2.2 LENGUAJE DE PROGRAMACIÓN

Se define como lenguaje de programación a un idioma artificial creado para posibilitar la comunicación entre el hombre y la computadora permitiendo la ejecución de instrucciones que procesan datos desde y hacia una base de datos. [3]

Se constituye por una cadena de símbolos y reglas sintácticas que delimitan su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, se prueba, se depura, se compila y se mantiene el código fuente de un programa informático se le llama programación.

2.2.1 Tipos de Lenguajes de Programación

Existen dos tipos de lenguajes claramente diferenciados; los lenguajes de bajo nivel y los de alto nivel.

El computador entiende un lenguaje conocido como código binario o código máquina, formado por ceros y unos. Es decir, sólo utiliza cero y uno para codificar cualquier acción.

Los lenguajes más próximos a la arquitectura hardware se denominan lenguajes de bajo nivel y los que se encuentran más cercanos a los programadores y usuarios se denominan lenguajes de alto nivel. [2]

LENGUAJES DE BAJO NIVEL.- Lenguajes dependientes de la máquina, el programa que se realiza con este tipo de lenguajes no se puede migrar o utilizar en otras máquinas. Al estar prácticamente diseñados a medida del hardware, aprovechan al máximo las características del mismo. [4]

El lenguaje máquina ordena las operaciones fundamentales para su funcionamiento. Consiste en la combinación de ceros y unos para formar las órdenes entendibles por el hardware de la máquina. Este lenguaje es mucho más rápido que los lenguajes de alto nivel. La desventaja es que son bastantes difíciles de manejar y usar, además de tener códigos fuente enormes donde encontrar un fallo es casi imposible. El lenguaje ensamblador es un derivado del lenguaje máquina y está formado por abreviaturas de letras y números llamadas mnemotécnicos.

Con la aparición de este lenguaje se crearon los programas traductores para poder pasar los programas escritos en lenguaje ensamblador a lenguaje máquina. Como ventaja con respecto al código máquina es que los códigos fuentes eran más cortos y los programas creados ocupaban menos memoria. Las desventajas de este lenguaje siguen siendo prácticamente las mismas que las del lenguaje ensamblador, añadiendo la dificultad de tener que aprender un nuevo lenguaje difícil de probar y mantener.

LENGUAJES DE ALTO NIVEL.- Son aquellos que se encuentran más cercanos al lenguaje natural entendible por el programador. Se trata de lenguajes independientes de la arquitectura del ordenador, por lo que, en principio, un programa escrito en un lenguaje de alto nivel, lo puedes migrar de una máquina a otra sin ningún tipo de problema. [4]

Estos lenguajes permiten al programador olvidarse por completo del funcionamiento interno de la máquina/s para la que están diseñando el programa. Tan solo necesitan un traductor o compilador que entiendan el código fuente y lo interprete en lenguaje máquina.

2.2.2 Lenguajes de Programación más Utilizados.

JAVA.- Es un lenguaje de programación orientado a objetos, desarrollado por Sun Microsystems a principios de los años 90. El lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria.

Con respecto a la memoria, su gestión no es un problema ya que ésta es gestionada por el propio lenguaje y no por el programador. [5]

C.- Es un lenguaje de programación creado en 1972 por Dennis M. Ritchie en los Laboratorios Bell como evolución del anterior lenguaje B, a su vez basado en BCPL. Al igual que B, es un lenguaje orientado a la implementación de Sistemas Operativos, concretamente Unix. C es apreciado por la eficiencia del código que produce y es el lenguaje de programación más popular para crear software de sistemas, aunque también se utiliza para crear aplicaciones. [5]

Se trata de un lenguaje débilmente tipificado de medio nivel pero con muchas características de bajo nivel. Dispone de las estructuras típicas de los lenguajes de alto nivel pero, a su vez, dispone de construcciones del lenguaje que permiten un control a muy bajo nivel. Los compiladores suelen ofrecer extensiones al lenguaje que posibilitan mezclar código en ensamblador con código C o acceder directamente a memoria o dispositivos periféricos. [5]

C++.- Lenguaje de programación diseñado a mediados de los años 1980 por Bjarne Stroustrup. El propósito de su creación fue el ampliarse al exitoso lenguaje de programación C con elementos que permitan la manipulación de objetos. En ese sentido, desde el punto de vista de los lenguajes orientados a objetos, el C++ es un lenguaje híbrido. Seguidamente se aumentaron facilidades de programación genérica, que se incorporó a los otros dos prototipos que ya estaban permitidos (programación estructurada y la programación orientada a

objetos). Una característica del C++ es la posibilidad de rediseñar los operadores (sobrecarga de operadores), y de poder crear nuevos tipos que se permitan como tipos fundamentales. [5]

C#.- Es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET. Es uno de los lenguajes de programación diseñados para la infraestructura de lenguaje común. Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma .NET, similar al de Java, aunque incluye mejoras derivadas de otros lenguajes. Existen muchos lenguajes de Programación, sin embargo se han referenciado los más utilizados por desarrolladores a nivel mundial mundo. [5]

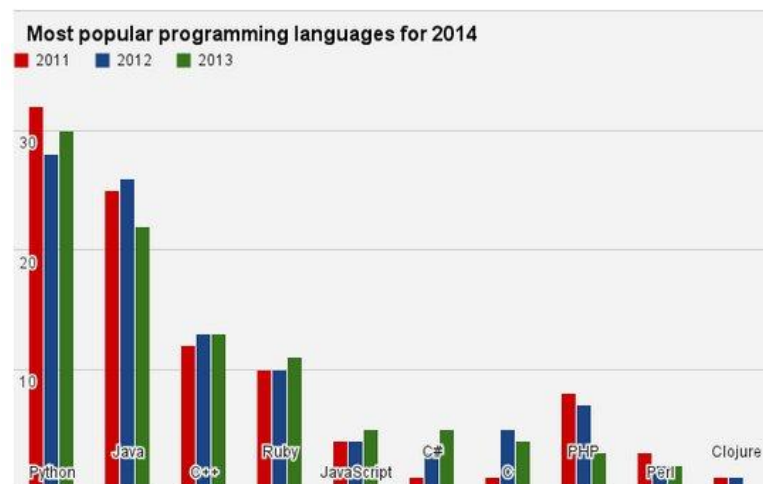


FIGURA 2. 1.- PRINCIPALES LENGUAJES DE PROGRAMACIÓN.

En Codeeval^[1] han hecho un estudio relacionado con los lenguajes de programación más utilizados y su evolución durante los años 2011, 2012 y 2013, números que venturebeat.com ha utilizado para crear la figura anterior.

2.3 MÉTRICAS DEL SOFTWARE

En el campo de la informática una métrica es cualquier medida o conjunto de medidas destinadas a conocer o estimar el tamaño u otra característica de un software o sistema de información, generalmente para realizar comparativas o para la planificación de proyectos de desarrollo o de actualizaciones oportunas que permitan corregir errores y ampliar el ciclo de vida de mencionado software. [6]

2.3.1 Factores de Calidad del Software

Los requisitos del Software son la base de las medidas de calidad. La falta de concordancia con los requisitos es una falta de calidad. Unos estándares específicos definen un conjunto de criterios de desarrollo que guían la manera en que se hace la ingeniería del Software. Si no se siguen los criterios, habrá seguramente poca calidad.

¹ **Coodeval**.- Comunidad para profesionales de la programación, con más de 24000 miembros que compiten entre ellos mostrando sus habilidades.

Existe un conjunto de requisitos implícitos que a menudo no se nombran. Si el software cumple con sus requisitos explícitos pero falla en los implícitos, la calidad del software no será fiable. [6]

2.3.2 Factores de calidad McCall

El modelo de McCall organiza los factores en tres ejes o puntos de vista desde los cuales el usuario puede contemplar la calidad de un producto, basándose en once factores de calidad organizados en torno a los tres ejes y a su vez cada factor se desglosa en otros criterios: [6]

TABLA 1 FACTORES DE CALIDAD MCCALL

Punto de Vista	Factores
Operación del producto	Facilidad de Uso, Integridad, Corrección, Fiabilidad y Eficiencia
Revisión del producto	Facilidad de Mantenimiento Facilidad de prueba Flexibilidad
Transición del producto	Facilidad de reutilización Interoperabilidad Portabilidad

TABLA 2 RELACIÓN DE FACTORES DE CALIDAD

Factor	Beneficio / Costo
Corrección	alto
Fiabilidad	alto
Eficiencia	bajo
Integridad	bajo
Facilidad de uso	medio
Facilidad de mantenimiento	alto
Facilidad de prueba	alto
Flexibilidad	medio
Portabilidad	medio
Reusabilidad	medio
Interoperabilidad	bajo

La mayoría de aplicaciones contienen errores en su lógica de programación debido en su mayor parte a la desorganización y falta de planificación, lo que conlleva a trabajar hoy corrigiendo lo que se hizo mal ayer. Esto influye en relación beneficio/costo por lo que se requerirá una reingeniería a fin de optimizar mencionado software. A fin de optimizar la planificación y diseño del software debemos basarnos en tres ejes fundamentales y sus respectivos criterios:

- Características Operativas.
- Capacidad de soportar los cambios.
- Adaptabilidad a nuevos entornos.

Características Operativas:

Corrección.- ¿Hace lo que quiero?

Fiabilidad.- ¿Lo hace de forma fiable todo el tiempo?

Eficiencia.- ¿Se ejecutará en mi hardware de la mejor manera?

Seguridad e Integridad.- ¿Es seguro?

Facilidad de Uso.- ¿Está diseñado para ser usado en forma amigable?

Capacidad de soportar cambios:

Facilidad de mantenimiento. ¿Puedo corregirlo?

Flexibilidad.- ¿Puedo cambiarlo?

Facilidad de Prueba.- ¿Puedo probarlo?

Adaptabilidad a nuevos entornos:

Portabilidad.- ¿Podré usarlo en otra máquina?

Reusabilidad.- ¿Podré reutilizar alguna parte del software?

Interoperabilidad.- ¿Podré hacerlo interactuar con otro sistema?

2.4 MARCO DEL DESARROLLO WEB

Un framework o infraestructura digital, es una organización conceptual y tecnológica de soporte específico, normalmente con instrumento o módulos de software concretos, de acuerdo al cual otro proyecto de software puede ser más fácilmente definido y desarrollado. [7]

En resumen, un conjunto estandarizado de conceptos, prácticas y razones para orientar un tipo de problemática personal, que sirve como dato de referencia para desafiar y resolver nuevos problemas de índole similar, incluyendo soporte de programas, bibliotecas, y un lenguaje interpretado, entre otros instrumentos, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Constituye una arquitectura de software que forma las relaciones generales de las entidades del dominio, y proporciona una estructura y una exclusiva metodología de trabajo, la cual ocupa o utiliza las aplicaciones del dominio.

2.4.1 Objetivos del Framework

El objetivo principal y general del framework es el desarrollo de un conjunto de herramientas y técnicas de programación que permitan implementar un framework de desarrollo de software de próxima generación (entendiendo a software de próxima generación como la

posibilidad de desarrollar software más abstracto, reutilizable e independiente de la plataforma). [7]

A partir de esto, podemos definir objetivos más específicos como:

- ✓ Permitir la utilización de toda la infraestructura existente en cada plataforma (bibliotecas de clases, componentes, etc.).
- ✓ Extender el tiempo de vida de una implementación a décadas (20, 30, 50 años).
- ✓ Reflexión completa, orientada a objetos, en tiempo de compilación.
- ✓ Desarrollo de software multiplataforma.
- ✓ Portabilidad entre plataformas sin pérdida de rendimiento.
- ✓ Permitir implementar el software desarrollado en plataformas de software o hardware inexistentes al momento del desarrollo inicial.
- ✓ Componentes modulares y abiertos (compilador extensible con "Plugins").
- ✓ Múltiples lenguajes de alto nivel como origen.
- ✓ Fácil implementación de lenguajes de alto nivel.
- ✓ Capacidades RAD incorporadas a la plataforma de desarrollo (compiladores) independientes del lenguaje de programación, entorno y plataforma.
- ✓ Lenguajes de alto nivel extensibles.
- ✓ Extensiones de lenguajes programables fácilmente y utilizables por cualquier lenguaje cliente.

2.4.2 Arquitectura

En este aspecto, nos fundamentamos en el modelo MVC (Controlador - Modelo - Vista), ya que debemos fragmentar nuestra programación y observar estos aspectos básicos en cuanto a la implementación de nuestro sistema: [8]

Controlador.- Podemos examinar el acceso a nuestra aplicación, y esto puede incluir: archivos, scripts, y/o programas; cualquier tipo de información que permita la conexión.

Así, podremos variar nuestro contenido de forma eficiente y estática; sólo debemos controlar ciertos aspectos.

Modelo.- Esta parte del MVC es la capa para acceder a datos desde la BDD.

Vista.- Finalmente, a este componente le corresponde dibujar, o expresar la última forma de los datos: la conexión gráfica que interactúa con el usuario final del programa.

TABLA 3 PRINCIPALES FRAMEWORKS PHP

Web Application Frameworks	
RUBY ON RAILS	Framework MVC basado en Ruby, orientado al desarrollo de aplicaciones web
CODELGNITER	Poderoso framework PHP liviano y rápido
KOHANA	Un fork de CodeIgniter
DJANGO	Framework Python que promueve el desarrollo rápido y el diseño limpio
CAKEPHP	Framework MVC para PHP de desarrollo rápido
ZEND FRAMEWORK	Framework para PHP 5, simple, claro y open-source
YII	Framework PHP de alto rendimiento basado en componentes
PYLONS	Framework web para Python que enfatiza la flexibilidad y el desarrollo rápido
CATALYST	Framework para aplicaciones web MVC elegante
SYMFONY	Framework full-stack
TURBOGEARS	Próxima generación construido sobre Pylons

2.4.3 Listado de Principales Frameworks PHP

Los mejores frameworks son especialmente buenos para organizar proyectos de gran magnitud, y a su vez tratando de mantenerse fuera del camino, sin imponerse por sobre el proyecto. El más conocido patrón de diseño de aplicaciones web es la arquitectura Model-View-Controller (MVC: Modelo-Vista-Controlador). [9]

Un Modelo es un objeto que representa datos o incluso una actividad en forma de tabla en base de datos. Una Vista es algún tipo de visualización del estado del modelo, y un Controlador ofrece funciones (herramientas) para cambiar el estado del modelo.

Hay una amplia gama de frameworks para aplicaciones web disponibles para Linux que son distribuidos bajo licencia Open Source. Para proporcionar una visión en el software que actualmente está disponible, hemos compilado una lista de 10 frameworks de alta calidad. Los que hemos seleccionado cubren un número de lenguajes de programación incluyendo Ruby, PHP y Python.

2.5 METODOLOGÍA RUP (RATIONAL UNIFIED PROCESS)

Es un proceso de Ingeniería del Software. Facilita una proximidad cuidadosa a la asignación de tareas y responsabilidades en una organización de

desarrollo. Su finalidad es fortalecer la producción de software de alta calidad que se ajuste a los requerimientos de sus beneficiarios finales con unos costos y calendario previsible. [10]

Este RUP permite optimizar la planificación del desarrollo de una aplicación aumentando la proyección de englobar mayor cantidad de procesos que conlleven a la elaboración de grandes proyectos de software. El RUP se basa en seis principios básicos como son:

Adaptar el proceso.- El proceso deberá adaptarse a las necesidades del cliente ya que es muy importante interactuar con él. Las características propias del proyecto u organización.

El tamaño del mismo, así como su tipo o las regulaciones que lo condicionen, influirán en su diseño específico. También se deberá tener en cuenta el alcance del proyecto en un área subformal. Aquí se analiza el alcance que tendrá la implementación del proyecto adaptando todas las actividades actuales tanto en la gestión de bodega como en la gestión de los recursos humanos de DIRGES.

Equilibrar prioridades.- Los requisitos de los diversos participantes pueden ser diferentes, contradictorios o disputarse recursos limitados. Debe encontrarse un equilibrio que satisfaga los deseos de todos. Gracias a este equilibrio se podrán corregir desacuerdos que surjan en el futuro.

Demostrar valor iterativamente.- Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los involucrados, tanto usuarios finales como administradores y directivos, la calidad y permanencia del producto, y se refina la dirección del proyecto así como también los riesgos involucrados.

Colaboración entre equipos.- El desarrollo de software no lo hace una única persona sino múltiples equipos. Debe haber una comunicación fluida para coordinar requisitos, desarrollo, evaluaciones, planes, resultados, etc.

Elevar el nivel de abstracción.- Este principio dominante motiva el uso de conocimientos reutilizables tales como patrón del software, lenguajes 4GL o marcos de referencia (frameworks) por mencionar algunos.

Esto impide que los ingenieros de software vayan directamente de las exigencias a la simbolización de software a la medida del cliente, sin saber con certeza qué codificar para satisfacer de la mejor manera los requisitos y sin comenzar desde un principio pensando en la reutilización del código. Un alto nivel de abstracción también permite discusiones sobre diversos niveles y soluciones arquitectónicas.

Éstas se pueden acompañar por las formas visuales de la arquitectura, por ejemplo con el lenguaje UML.

Enfocarse en la calidad.- El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción. El aseguramiento de la calidad forma parte del proceso de desarrollo y no de un grupo independiente.

2.5.1 Características

Maneja casos de uso.- La misión de un buen sistema es lograr satisfacer las necesidades para lo que fue concebido, pudiendo interactuar con usuarios humanos y a su vez brindar información a otros sistemas para proyectos posteriores o procesos relacionados. Los casos de uso recrean de manera óptima los diferentes procesos que el sistema automatizará, y que a su vez reflejará la efectividad del mismo. Se podrá evidenciar los beneficios que el sistema brindará en el tratamiento de la información recibida y posteriormente entregada.

Basado en Arquitectura.- Comprende los elementos más específicos del sistema y está influenciada entre otros por plataformas de software, sistemas operativos, manejadores de bases de datos, protocolos, consideraciones de desarrollo como sistemas adquiridos y exigencias no funcionales.

Es como una placa del sistema que estamos desarrollando, lo suficiente completa como para que todos los involucrados en el desarrollo tengan una idea clara de qué es lo que están cimentando. Se representa mediante varias vistas que se centran en aspectos concretos.

Iterativo e Incremental: Para hacer más adaptable un proyecto se recomienda dividirlo en ciclos. Para cada ciclo se implantan fases de referencia, cada una de las cuales debe ser considerada como un mini proyecto cuya sustancia fundamental está establecida por una o más iteraciones de las actividades importantes básicas de cualquier proceso de desarrollo. En concreto RUP divide el proceso en cuatro etapas, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto.

2.5.2 Fases de la Metodología

Inicio.- Antes de iniciar un proyecto es beneficioso plantearse algunas cuestiones: ¿Cuál es el objetivo? ¿Es factible? ¿Lo construimos o lo compramos? ¿Cuánto va a costar? La etapa de inicio trata de responder a estas preguntas y a otras más. Sin embargo no intentamos una estimación precisa o la captura de todas las exigencias. Generalmente no debe durar mucho más de una semana definir los objetivos necesarios como:

- ✓ Establecer el ámbito del proyecto y sus metas.
- ✓ Situar los casos de uso críticos del sistema, los espacios básicos que definen la funcionalidad.
- ✓ Expresar al menos una arquitectura candidata para los espacios principales.
- ✓ Evaluar el coste en recursos y tiempo de todo el proyecto.
- ✓ Considerar los riesgos, las fuentes de indecisión.

Al finalizar la fase de inicio se deben exponer los criterios de evaluación para continuar:

- ✓ Todos los comprometidos en el proyecto concuerdan en la definición del ambiente del sistema y las valoraciones de agenda.
- ✓ Entendimiento de los requerimientos, demostrado por la fidelidad de los casos de uso principales.
- ✓ Las estimaciones de tiempo, coste y riesgo son ciertos.
- ✓ Conocimiento total de cualquier esquema de la arquitectura desarrollado.
- ✓ Los gastos hasta el momento se parecen a los proyectados.

Si el proyecto no pasa estos criterios hay que proyectarse renunciarlo o estudiarlo profundamente.

Elaboración.- La intención de esta fase es examinar el dominio del problema, implantar los cimientos de la arquitectura, desarrollar el plan del proyecto y relegar los mayores conflictos. Cuando concluye esta fase se llega al punto de no retorno del proyecto: a partir de ese momento pasamos de las relativamente ligeras y de poco conflicto dos primeras fases, a afrontar la etapa de construcción, costosa y arriesgada.

Es por esto que ésta es de gran relevancia. Se cimenta un prototipo de la arquitectura, que debe desarrollarse en iteraciones continuadas hasta convertirse en el sistema final. Debe contener los casos de uso críticos identificados en la etapa de inicio. También debe demostrarse que se han evitado los conflictos más graves, bien con este prototipo, bien con otros de usar y tirar.

Los objetivos de esta fase son:

- ✓ Definir, autorizar e implantar la arquitectura.
- ✓ Completar la visión.
- ✓ Establecer un plan íntegro para la etapa de construcción. Este plan puede desarrollarse en repetidas iteraciones. Debe incluir los costes si procede.
- ✓ Expresar que la arquitectura propuesta resistirá la visión con un coste moderado y en un tiempo razonable.

Al concluir deben alcanzarse los siguientes productos:

- ✓ Un modelo de casos de uso completo al menos hasta el 80%: todos los casos y actores identificados, la mayoría de los casos desarrollados.
- ✓ Requerimientos agregados.
- ✓ Representación de la arquitectura software.
- ✓ Un modelo permitido de la arquitectura.
- ✓ Caso de negocio examinados y lista de conflictos.
- ✓ Plan de mejora para el proyecto.
- ✓ Un caso de desarrollo actualizado que define el proceso a seguir.
- ✓ Posiblemente un manual de usuario preliminar.
- ✓ La forma de aproximarse a esta fase debe ser tratar de englobar todo el proyecto con la profundidad mínima. Sólo se agudiza en los puntos críticos de la arquitectura o conflictos importantes.

En la presente etapa de elaboración se sustituyen todos los productos de la fase de inicio como el glosario y el caso de negocio. Las consideraciones de valoración de esta fase son los siguientes:

- ✓ La visión del producto es constante.
- ✓ Es estable la arquitectura.

- ✓ Se ha expuesto mediante la ejecución del prototipo que los principales elementos de riesgo han sido abordados y resueltos.
- ✓ El propósito para la fase de construcción es minucioso
- ✓ y preciso. Las estimaciones son fidedignas.
- ✓ Todos los comprometidos armonizan en que la visión actual será lograda si se siguen los procedimientos actuales en el contexto de la arquitectura actual.
- ✓ Los gastos hasta ahora son aceptables, comparados con los previstos.
- ✓ Si no se culminan los criterios de evaluación quizá sea preciso dejar el proyecto o modificarlo considerablemente.

Construcción.- El propósito primordial de esta fase es lograr la capacidad operacional del producto de forma incremental a través de las continuadas iteraciones. Aquí todos los mecanismos, características y exigencias, que no hayan sido hecho hasta ahora, han de ser efectuados, integrados y testeados, alcanzando una versión del producto que se pueda poner en manos de los beneficiarios (una versión beta). La importancia en esta fase puede controlar las operaciones realizadas, gestionando los recursos eficientemente, de tal forma que se mejoren los costes, los calendarios y la calidad.

Los objetivos concretos incluyen:

- ✓ Disminuir los costes de desarrollo mediante la optimización de recursos y evadiendo el tener que rehacer un trabajo o incluso descartarlo.
- ✓ Obtener una calidad apropiada tan rápido como sea práctico.
- ✓ Alcanzar versiones funcionales (alfa, beta, y otras versiones de prueba) tan rápido como sea práctico.

Los productos en la fase de construcción deben ser:

- ✓ Modelos Completos (Casos de Uso, Análisis, Diseño, Despliegue e Implementación).
- ✓ Arquitectura íntegra (conservada y mínimamente renovada).
- ✓ Conflictos Demostrados Atenuados.
- ✓ Plan del Proyecto para la fase de Transición.
- ✓ Manual Inicial de Usuario (con suficiente detalle).
- ✓ Prototipo Operacional – beta.
- ✓ Caso del Negocio Actualizado.

Transición.- El objetivo de esta fase es ubicar el producto en manos de los beneficiarios finales, para lo que típicamente se demandará desarrollar nuevas versiones actualizadas del producto, completar la documentación, instruir al usuario en el manejo del producto, y en general tareas relacionadas con el ajuste, configuración, instalación y usabilidad del producto.

En definitiva se mencionan algunas de las cosas que puede incluir esta fase:

- ✓ Testeo de la versión Beta frente a las posibilidades de los usuarios para validar el nuevo sistema.
- ✓ Funcionamiento paralelo con los sistemas legados que están siendo reemplazados por nuestro proyecto.
- ✓ Conversión de las bases de datos operacionales.
- ✓ Preparación de los usuarios y técnicos de mantenimiento.
- ✓ Entrega del producto a los equipos de marketing, distribución y venta.

Los principales objetivos de esta fase son:

- ✓ Obtener que el beneficiario se valga por sí mismo.
- ✓ Un producto final que cumpla con los requerimientos esperados, que funcione y satisfaga suficientemente al usuario.

Los productos de la fase de transición son:

- ✓ Prototipo Operacional.
- ✓ Documentos Legales.
- ✓ Caso del Negocio Completo.
- ✓ Línea de Base del Producto completa y corregida que contiene todos los modelos del sistema.

- ✓ Descripción de la Arquitectura completa y corregida.

Las iteraciones de esta etapa irán dirigidas normalmente a obtener una nueva versión. Las diligencias a efectuar durante las iteraciones dependerán de su objetivo, si es corregir algún error detectado, regularmente será suficiente con llevar a cabo los flujos de trabajo de implementación y test, sin embargo, si se deben añadir nuevas características.

La iteración será equivalente a la de una iteración de la fase de construcción. La complejidad de esta fase obedece totalmente de la naturaleza del proyecto, de su alcance y de la organización en la que deba implantarse.

2.5.3 Ciclo de Vida

En el ciclo de vida del RUP se observa una implementación de desarrollo en espiral. Con el ciclo de vida se establecen tareas en fases e iteraciones. El RUP opera el proceso en cuatro etapas, dentro de las cuales se realizan varias iteraciones en número variable. [11]

Las primeras iteraciones (en las fases de Inicio y Elaboración) se orientan hacia la comprensión del problema y la tecnología, la

delimitación del ámbito del proyecto, la eliminación de los riesgos críticos, y al establecimiento de una base de inicio.

El RUP en cada una de sus fases realiza una serie de artefactos que sirven para comprender mejor tanto el análisis como el diseño del sistema estos artefactos son los siguientes:

Inicio:

- + Documento Visión.
- + Especificación de Requerimientos.

Elaboración:

- + Diagramas de caso de uso.

Construcción:

- + Documento Arquitectura que trabaja con las siguientes vistas:

Vista Lógica:

- + Diagrama de clases.
- + Modelo E-R (Si el sistema así lo requiere).

Vista de Implementación:

- + Diagrama de Secuencia.
- + Diagrama de estados.
- + Diagrama de Colaboración.

Vista Conceptual:

✚ Modelo de dominio.

Vista física:

✚ Mapa de comportamiento a nivel de hardware.

2.6 INVENTARIOS

El inventario es el registro documental de los bienes y demás objetos pertenecientes a una persona física, a una comunidad y que se haya realizado a partir de mucha exactitud y prolijidad en la plasmación de los datos. [12]

Igualmente y como resultado de la situación recién indicada, se llama inventario a la comprobación y recuento, tanto cualitativo como cuantitativo de las existencias físicas con las teóricas que fueron oportunamente justificadas.

En la gestión empresarial, lo que hace el inventario es registrar el conjunto de los bienes propios y disponibles de una empresa para la venta a sus clientes y que por tanto son considerados como activo corriente. Los bienes que son plausibles de ser sometidos a inventario se localizan destinados a la venta directa o a aquellas actividades destinadas de forma interna al proceso productivo, como es el caso de las materias primas, los productos no

terminados, las piezas de recambio destinadas a mantenimiento, materiales de embalaje, bienes de la empresa, bienes terminados, bienes parcialmente terminados, bienes en tránsito y materiales de envasado, entre otros. Existe una nutrida cantidad de tipos de inventarios, entre los más demandantes se cuentan:

- ✓ Inventarios Finales.- Se lleva a cabo cada vez que se cierra el período fiscal, generalmente el 31 de diciembre.
- ✓ Inventarios Periódicos.- Se realiza cada determinado tiempo.
- ✓ Inventarios Iniciales.- Se registran todos los bienes de la empresa.
- ✓ Inventarios de Liquidación Legal.
- ✓ Inventario de materias primas.
- ✓ Inventario de seguridad.
- ✓ Inventario de gestión.
- ✓ Inventario Físico, entre otros.

Entre los motivos que existen en las empresas para efectuar un inventario podemos citar: para reducir costos de adquisición, para reducir costos de calidad por arranque, para reducir costos vinculados al material faltante y para reducir costos de pedidos.

Los procesos de compras e inventario se realizan de la siguiente forma: El Pañol de consumos o BODEGA, se encarga de controlar y administrar el inventario de consumo interno y el inventario para el suministro general de los retenes o destacamentos pertenecientes a la DIRGES. Ambos tipos de

inventario son almacenados por separado. Cada departamento de los distintos retenes genera la solicitud de materiales que les son necesarias para su funcionamiento.

Este documento es registrado por el personal de BODEGA para su posterior despacho. BODEGA realiza el despacho de las Solicitudes de Materiales e insumos dependiendo de si hay las existencias de la mercancía solicitada por los retenes de su dependencia.

Basado en un punto de reorden (de cada uno de los artículos), se realizan reportes trimestrales de los artículos del inventario que deben ser comprados por la DIRGES. Cuando se agota cualquiera de los inventarios, BODEGA genera un documento llamado Solicitud de Requerimiento de Mercancía, donde colocan todos aquellos artículos que es necesario comprar. El departamento de COMPRAS PÚBLICAS recibe y registra todas las solicitudes de Requerimientos de Mercancía. Se compra dicha mercancía dependiendo de la disposición de partidas presupuestarias y del monto de los procesos, pudiendo ser por compra de ínfima cuantía o por subasta inversa, mediante el portal de compras públicas del INCOP.

CAPÍTULO 3

ANÁLISIS PARA LA IMPLEMENTACIÓN DEL SISTEMA INFORMÁTICO

3.1 ELECCIÓN DE HERRAMIENTAS DE DESARROLLO Y GESTOR DE BASE DE DATOS

A fin de estudiar las herramientas de desarrollo de software, analizaremos y evaluaremos los elementos fundamentales de una aplicación web como son:

- ✓ Plataforma Operativa.
- ✓ Servidor Web.
- ✓ Lenguaje de Desarrollo.
- ✓ Estándares de Desarrollo.
- ✓ Herramienta de la Base de Datos a utilizar.

Posteriormente, analizamos las características técnicas que se tomarán en cuenta en la elaboración del sistema, comparando las alternativas del uso del software (libre o propietario) que son las siguientes:

- ✓ Requerimientos de Hardware y Software tanto para el servidor como para el cliente.
- ✓ Costos operativos de tecnología.
- ✓ Compatibilidad con la plataforma tecnológica existente o proyectada.
- ✓ Estabilidad, fiabilidad y facilidad de uso.
- ✓ Seguridad.
- ✓ Documentación.
- ✓ Existencia de implementaciones de software de éxito comprobadas en el ámbito local y nacional.
- ✓ Disponibilidad de las actualizaciones del software.
- ✓ Los costos de las herramientas a utilizar.

De acuerdo al decreto 1014 emitido por parte de la presidencia del Ec. Rafael Correa Delgado que promueve el uso de software libre en las instituciones públicas del Ecuador, se decide utilizar las siguientes tecnologías:

- ✓ Servidor de Aplicaciones Web Apache.
- ✓ Lenguaje PHP.
- ✓ Motor de Base de Datos MySQL.
- ✓ El sistema operativo del servidor quedará abierto entre Linux y Windows ya que no afectaría el funcionamiento de la aplicación.
- ✓ Navegador Mozilla Firefox o Google Chrome en sus versiones más recientes.

3.1.1 Servidor de Aplicaciones Web Apache

Un servidor apache es un servidor web de distribución libre y de código abierto, siendo el más popular del mundo desde abril de 1996, con una penetración actual del 50% del total de servidores web del mundo (agosto de 2013). La principal competencia de Apache es el IIS (Microsoft Internet Information Services) de Microsoft. [13]

Apache fue la primera alternativa viable para el servidor web de Netscape Communications, actualmente conocido como Sun Java System Web Server. Apache es desarrollado y mantenido por una comunidad abierta de desarrolladores bajo el auspicio de la Apache Software Foundation. La aplicación permite ejecutarse en múltiples sistemas operativos como Windows, Novell NetWare, Mac OS X y los sistemas basados en Unix.

Características de Apache:

- ✓ Soporte para los lenguajes perl, python, tcl y PHP.
- ✓ Módulos de autenticación: mod_access, mod_auth y mod_digest.
- ✓ Soporte para SSL y TLS.
- ✓ Permite la configuración de mensajes de errores personalizados y negociación de contenido.
- ✓ Permite autenticación de base de datos basada en SGBD.

Uso de Apache:

Principalmente usado para servir páginas web estáticas y dinámicas en la WWW. Apache es el servidor web del popular sistema XAMP, junto con MySQL y los lenguajes de programación PHP/Perl/Python. La "X" puede ser la inicial de cualquier sistema operativo, si es Windows: WAMP, si es el Linux: LAMP, etc.

3.1.2 PHP

Lenguaje de programación usado generalmente en la creación de contenidos para sitios web (PHP Hypertext Pre-processor). Es un lenguaje interpretado especialmente usado para crear contenido dinámico web y aplicaciones para servidores, aunque también es posible crear aplicaciones gráficas utilizando la biblioteca GTK+. Generalmente los scripts en PHP se embeben en otros códigos como HTML, ampliando las posibilidades del diseñador de páginas web enormemente. [14]

La interpretación y ejecución de los scripts PHP se hacen en el servidor, el cliente (un navegador que pide una página web) sólo recibe el resultado de la ejecución y jamás ve el código PHP. Permite la conexión a todo tipo de servidores de base de datos como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite. PHP es una alternativa a otros sistemas como el ASP.NET, C#, VB.NET de Microsoft o a ColdFusion de Macromedia, a JSP,

Java de Sun Microsystems, y a CGI/Perl. La ventaja con los de Microsoft o Macromedia es que es totalmente gratuito, no hay que pagar licencias.

3.1.3 Base de Datos MySQL

MySQL es un sistema de gestión de bases de datos (SGBD) multiusuario, multiplataforma y de código abierto. MySQL pertenece a la compañía sueca MySQL AB, a la que le pertenece casi todos los derechos del código fuente. La compañía desarrolla y mantiene el sistema, vendiendo soporte y servicios, como también las licencias para usar MySQL. [15]

Uso de MySQL.-

Muy popular en aplicaciones web, y es componente de las plataformas LAMP, MAMP, WAMP, entre otras. MySQL suele combinarse con el popular lenguaje PHP.

Características de MySQL.-

- ✓ MySQL está escrito en C y C++.
- ✓ Emplea el lenguaje SQL para consultas a la base de datos.
- ✓ MySQL Server está disponible como freeware bajo licencia GPL.
- ✓ MySQL Enterprise es la versión por suscripción para empresas, con soporte las 24 horas.

Plataformas compatibles.-

Trabaja en las siguientes plataformas: AIX, BSDi, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell NetWare, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Microsoft Windows (95, 98, ME, NT, 2000, XP, Vista, 7, 8).

3.2 ANÁLISIS DE MÓDULOS DEL SISTEMA

El portal web de la DIRGES, cuenta con una página de Servicios, el mismo que identifica claramente dos módulos como se describe en la figura siguiente:

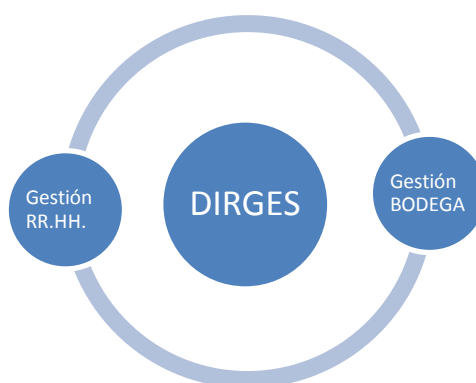


FIGURA 3.1 MÓDULOS DEL SISTEMA WEB DIRGES

El sistema tendrá la posibilidad de abrir cada módulo con su mismo usuario, depende de los roles que le asigne el Administrador de sistema. Esto se debe a que no todos los usuarios del sistema pueden hacer pedidos a

Bodega, generalmente lo hace el encargado de suministros de cada reparto militar o el personal de mayor grado de antigüedad para efectos de registro de actas de descargo del material.

3.2.1 Módulo de Gestión Recursos Humanos

El módulo de Gestión de Recursos Humanos comprende el manejo de toda la información personal y laboral del personal que labora en la DIRGES y sus retenes navales subordinados.

El objetivo principal de éste módulo es automatizar el proceso de asignación de licencias y permisos del personal y la distribución del mismo conforme la estructura orgánica del reparto. A continuación se presenta un esquema de las tareas principales que maneja el módulo de recursos humanos.



FIGURA 3.2 MÓDULO DE GESTION DE RECURSOS HUMANOS

Estructura Orgánica.- Esta opción permite administrar la estructura orgánica de la DIRGES con sus departamentos y divisiones correspondientes.

Esto facilita la ubicación específica del recurso humano en cada una de las oficinas de la organización. Todos los departamentos son organizados jerárquicamente, lo que facilita la relación de dependencia entre departamentos y divisiones.

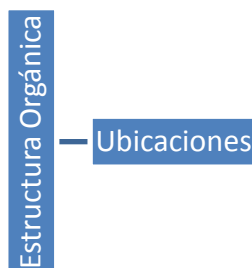


FIGURA 3.3 MÓDULO DE ESTRUCTURA ORGÁNICA ORGANIZACIONAL

Ubicaciones.- Dentro de la estructura de la DIRGES existen retenes navales en diferentes ciudades de la Provincia de Esmeraldas, por lo que en este módulo se pueden definir las ubicaciones o direcciones geográficas de los retenes navales de su jurisdicción.

Información del Personal.- En este sub módulo se mantendrá actualizada la información personal y laboral del recurso humano.

Para una mejor navegación, se ha agrupado la información en diferentes categorías:

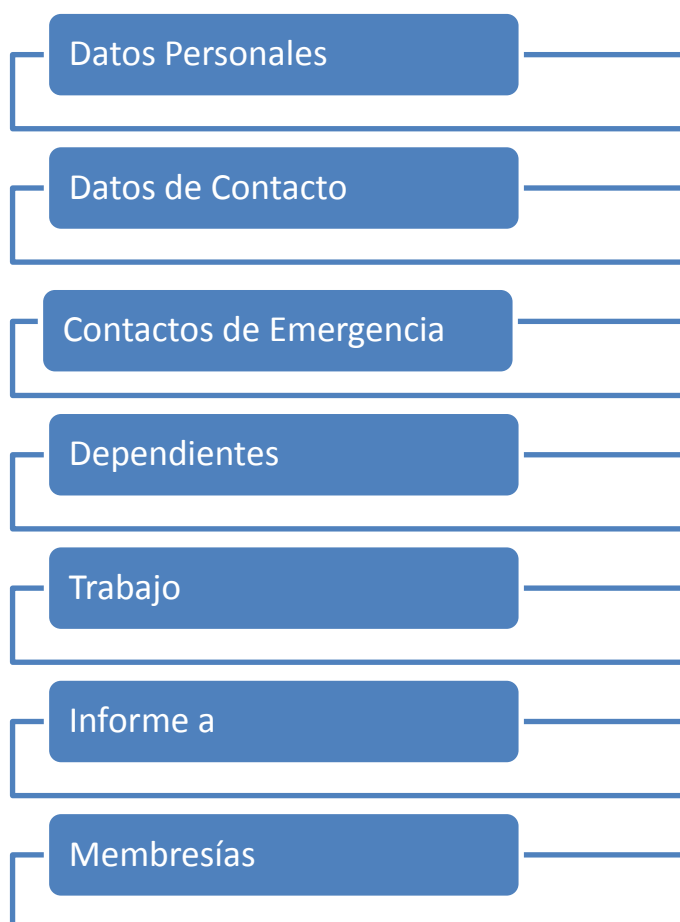


FIGURA 3.4 CATEGORÍAS PARA INFORMACIÓN PERSONAL DE EMPLEADOS

Datos Personales.- En esta categoría encontraremos los datos netamente personales de los empleados, tales como: nombres y apellidos completos, identificación, id o cédula, licencia de conducir,

genero, nacionalidad, estado civil, fecha de nacimiento, código ISSFA, especialidad y grado.

Datos de Contacto.- Se considerarán datos de contacto como: dirección 1, dirección 2, ciudad, provincia, país, teléfono de domicilio, teléfono móvil, teléfono del trabajo, email del trabajo, email particular.

Contactos de Emergencia.- Se guarda información de una persona de contacto en caso de emergencia tales como: nombre, relación, teléfonos de domicilio, móvil y del trabajo. Pueden almacenarse varios contactos de emergencia de ser el caso.

Dependientes.- Se guardan datos de los dependientes del empleado tales como: nombres, relación y fecha de nacimiento.

Trabajo.- Se almacena información netamente laboral tales como: título del trabajo, trabajo específico, estado del empleo, categoría del trabajo, fecha de registro, subunidad asignada, ubicación, fecha de ingreso a la armada.

De igual manera se puede adjuntar archivos del contrato de ser el caso en formato pdf.

Los datos de título de trabajo, estado del empleo, categoría del trabajo, subunidad y ubicación son datos pre configurados por el

administrador del sistema y presentados en listas desplegables para su selección.

Informe.- Se almacena la relación de dependencia de los empleados, debido a la jerarquización militar existente en la DIRGES. Los supervisores son seleccionados de la base de datos de empleados registrados en el sistema.

Membresías.- Permite registrar las membresías de los empleados a instituciones como: cooperativas de ahorro, vivienda, casinos, clubs, etc., con sus correspondientes descuentos.

Licencias y Permisos.- En esta opción se realizarán todos los registros de licencias y permisos debidamente planificadas y aprobadas por el usuario supervisor del sistema.

Derechos de Licencia.- Se podrá asignar derechos a licencias para los empleados de un departamento, o para un empleado en particular tomando en cuenta que por la particularidad del trabajo militar existen días de compensación que pueden ser acumulados como días de licencia para su posterior uso cuando el empleado así lo programe.

Informes.- Se podrá contar con un informe consolidado de todo el personal de la DIRGES con un balance de días tomados y días pendientes de vacaciones, esto facilitará la planificación de licencias sin afectar la operatividad de la organización.

Configuración.- Se podrá determinar los periodos de licencias de acuerdo a los días calendarios, tipos de licencia como: licencia anual, licencia paternal, periodo de maternidad, calamidad doméstica, permisos, etc., cada uno de estas con el número de días que pueden ser tomados pudiendo o no ser descontables de los treinta días de licencia anual para todos los empleados.

De igual manera se puede determinar la semana laboral a considerarse y las vacaciones extras que pudieran considerarse.

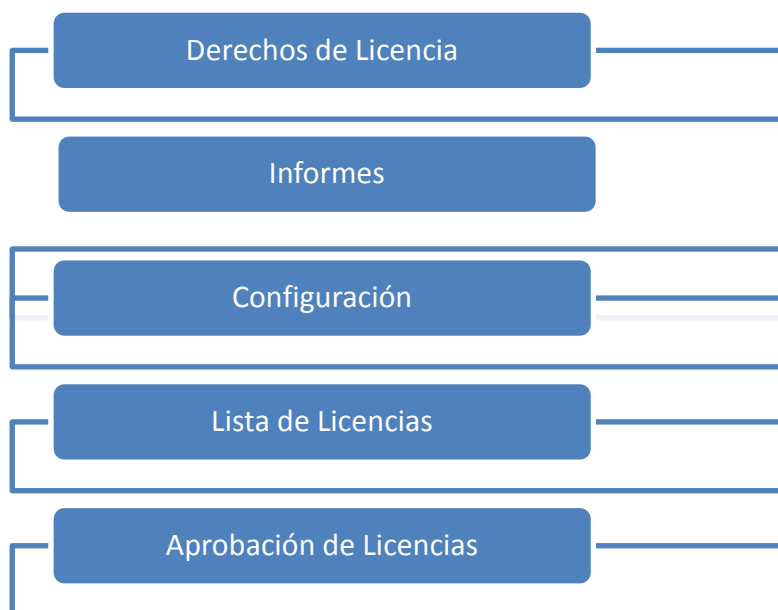


FIGURA 3.5 CATEGORÍAS PARA INFORMACIÓN RELACIONADA A LICENCIAS

Lista de Licencias.- Se puede obtener reportes en un rango de fechas determinado aplicando filtros de búsqueda como: licencias rechazadas, canceladas, pendientes de aprobación, programadas, tomadas, por empleado o por subunidad de la organización.

Aprobación de Licencias.- Se podrá gestionar la aprobación o rechazo de las licencias agendadas considerando siempre el número de personal efectivo que deberá laborar a fin de no afectar el cumplimiento de las funciones de la organización.

Tiempo y Asistencia.- Se podrá llevar un registro del control de asistencias y días de faltas del personal. La asistencia se podrá gestionar con el registro de la hora de ingreso y de la hora de salida del personal.

Reportes.- Se podrá contar con reportes oportunos de diferente índole, en caso de no haber un reporte predefinido, se podrá diseñar reportes personalizados mediante la selección de campos de datos de acuerdo a la necesidad del reporte.

3.2.2 Módulo de Gestión de Bodega

Para la Administración de Bodega se instalarán dos sistemas, uno de escritorio donde se ingresarán los productos y los catálogos con las cantidades de inventario, y otro sistema basado en la web donde se

gestionará su egreso y aprobación de los repartos, cabe mencionar que los dos sistemas son de código abierto.

Administración de Bodega.- Esta opción permitirá administrar el sitio de una manera amigable manejando los productos y un entorno gráfico y generando códigos de barra para su rápida identificación simplificado para su fácil uso.

Administración de Catálogo.- Admitirá manejar una lista dinámica de productos, administrarlo por categorías que permitan la rápida búsqueda de los productos y suministros dentro de toda la bodega.

Visualización de Productos.- Muestra los productos de forma única suministrando a los usuarios del sistema de opciones numerosas para visualizar los productos deseados. Esto permitirá:

- ✓ Comparar productos.
- ✓ Imágenes del producto.
- ✓ Búsqueda rápida de productos.
- ✓ Búsqueda por alias.
- ✓ Elección de número de productos a mostrar por página.
- ✓ Mostrar cantidades disponibles.
- ✓ Mostrar productos en la misma categoría.
- ✓ Construir atributos de productos.

- ✓ Marcar productos.
- ✓ Mostrar el contenido de cesta.
- ✓ Mostrar categorías de producto.
- ✓ Mostrar cantidades disponibles de producto en inventario.

Optimización de Motores de Búsqueda.- El mercadeo de motores de búsqueda es uno de los mejores tipos de mercadeo mediante la simple colocación de palabras claves que asocian a los productos existentes en bodega. Esto permite:

- ✓ Editar etiquetas de producto.
- ✓ Reducir el tiempo de carga de los productos.
- ✓ Prevenir contenido duplicado.

Finalización de Pedidos.- Una vez que el supervisor valide el requerimiento del reparto requirente, se aprobará la solicitud que será enviada al correo electrónico del solicitante y se devengará de los inventarios de la bodega.

Envíos.- Una vez finalizado se coordinará con el Departamento de Transporte el envío de los productos seleccionados, esto se lo coordinará vía correo electrónico con el requirente.

- ✓ Transportistas predefinidos (Choferes de reparto).
- ✓ Notificaciones por e-mail del envío.

- ✓ Recoger en la bodega de DIRGES.

Seguridad.- Este módulo permite brindar a los usuarios del sistema un ambiente de confianza en sus pedidos mediante un sistema de encriptación de contraseñas. Esto permite:

- ✓ Resistencia a los ataques de tipo XSS, CSRF, inyecciones SQL, inserciones distantes, recorrido transversal.
- ✓ Bloqueo a las inyecciones de encabezado de e-mail.
- ✓ Bloqueo de las tentativas de recuperación de contraseñas repetidas.

Informes.- Son la clave para el seguimiento y la optimización del rendimiento. Los directivos de la organización deben monitorear los pedidos y las interacciones de los usuarios del sistema para interpretar los requerimientos que se están solventando y lo que se necesite implementar, esto permitirá:

- ✓ Exportar listas formato Excel o pdf.
- ✓ Seguimiento a la actividad del usuario.
- ✓ Ver perfiles de usuarios.
- ✓ Informes de pedidos.
- ✓ Estadísticas de pedidos por usuario.
- ✓ Categorías y productos de mayor movimiento.
- ✓ Estadística de interacción de usuarios.

CAPÍTULO 4

DESARROLLO DEL APLICATIVO

4.1 PLANIFICACIÓN DEL SISTEMA

Una correcta planificación del proyecto para el desarrollo de un software facilitará un marco de trabajo óptimo, lo que permite a los programadores efectuar estimaciones razonables de recursos, costos y planificación temporal.

Estas estimaciones se formalizan dentro de un marco de tiempo limitado al inicio del proyecto de software, y deberían renovarse periódicamente a medida que progresa el proyecto.

Igualmente las estimaciones deberían precisar los escenarios del mejor y peor caso, de manera que los resultados del proyecto pueden limitarse, y ajustarse a los tiempos y plazos establecidos.

Esto se logra mediante la metodología de unificación de procesos, la misma que permite utilizar sus normas para definir el proyecto y de la misma manera permite establecer de mejor manera tanto para desarrollo como para documentación.

4.1.1 Propósito

El propósito para este proyecto es definir, planificar y controlar el desarrollo de un producto totalmente amigable y funcional en un periodo de tiempo específico, tomando en cuenta todas las variables del desarrollo del proyecto a fin de garantizar su funcionamiento y adaptabilidad a cambios futuros de acuerdo a requerimientos de actualizaciones que puedan surgir en la empresa.

4.1.2 Vista General del Sistema

Los módulos de Gestión de Recursos Humanos y de Gestión de Bodega se centrarán en una interface web bajo la denominación de Servicios en Línea, solo el módulo de registro de productos en bodega será plataforma cliente/servidor en java, y su base de datos sincronizará en la web para consultas y gestión de los retenes navales.

El acceso a la aplicación será mediante cualquier navegador web, por lo que facilitará el acceso remoto al mismo desde cualquier lugar que

se requiera mediante el acceso al internet. Esta aplicación permitirá tener registro digital de todas las tareas de gestión de recursos humanos y de movimientos de insumos en la bodega de la DIRGES.

Dentro del sistema se establece dos módulos claramente definidos e independientes en su propósito:

- ✓ Módulo de Gestión de Recursos Humanos.
- ✓ Módulo de Gestión de Bodega.

4.1.3 Suposiciones y Restricciones

Las suposiciones y restricciones respecto a la aplicación permiten identificar los siguientes puntos críticos:

- ✓ Seguridad en la aplicación.
- ✓ Velocidad de transferencia de información en los procesos del sistema.
- ✓ Manejo y adaptación del usuario directo del sistema.

4.1.4 Entregables del Proyecto

A continuación, se muestran y detallan cada uno de los instrumentos que serán generados y utilizados por el proyecto, estos establecen los artefactos entregables. Esta lista constituye la configuración o proyecto del RUP desde la perspectiva de artefactos el mismo que será utilizado

para el desarrollo de este plan. Es necesario enfatizar que de acuerdo a la filosofía de RUP (todo proceso es iterativo e incremental), todos los artefactos son centro de alteraciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso lograríamos tener una versión definitiva y completa de cada uno de ellos.

Sin embargo, el efecto de cada iteración y los hitos del proyecto están orientados a lograr un cierto grado de completitud y permanencia de los artefactos, siendo estos:

Plan de desarrollo del software.- Corresponde todo el análisis y definición final del proyecto, por lo que podemos deducir el presente documento.

Visión.- El presente documento define la perspectiva de los usuarios con respecto al proyecto, especificando las necesidades y características del mismo, y las bondades que el proyecto ofrecerá al final.

Glosario.- Permite establecer una terminología consensuada.

Especificaciones de casos de uso.- Es una descripción especificada utilizando una plantilla de documento, donde se incluyen:

precondiciones, post-condiciones, flujo de eventos, necesidades no-funcionales asociados.

Para casos de uso, cuyo flujo de eventos sea complicado podrá adherirse una representación gráfica mediante un diagrama de actividad.

Estos casos de uso segmentan el conjunto de necesidades atendiendo a la categoría del usuario que participa en el mismo, está basado en lenguaje natural, es decir, es más comprensible por los usuarios finales.

Prototipos de interfaces de usuario.- Son prototipos que permiten al usuario apreciar y familiarizarse con la interfaz que proporcionará la aplicación y así, obtener retroalimentación de su parte respecto a los exigencias del proyecto.

Estos prototipos se efectuarán con las mismas plantillas predefinidas con las que se realizará todo el proyecto obteniendo un producto en el que se pueda navegar y ejecutar la aplicación.

Así mismo, este artefacto, será rechazado en la etapa de construcción en la medida que el resultado de las iteraciones vayan desarrollando el producto final.

Modelo de análisis y diseño.- Establece la elaboración de los casos de uso en clases y pasando desde una representación en términos de análisis (sin incluir aspectos de implementación) hacia una de diseño (incluyendo una orientación hacia el entorno de implementación), de acuerdo al progreso del proyecto.

Modelo de datos.- Este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos, pronosticando que la persistencia de la información del sistema será soportada por una base de datos relacional.

Modelo de implementación.- Es una compilación de elementos y los subsistemas que los contienen. Estos elementos incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para el establecimiento y desarrollo del sistema.

(Este modelo es sólo una versión preliminar al final de la fase de elaboración, posteriormente tiene bastante refinamiento conforme a las exigencias que vayan surgiendo).

Manual de instalación.- Incluye las instrucciones para efectuar la instalación del producto.

Producto.- La estructura del sistema DIRGES será empaquetada y almacenada en un CD con los componentes adecuados para facilitar su instalación. El producto, a partir de la primera iteración de la fase de construcción es desarrollado incremental e iterativamente, alcanzando una nueva versión al final de cada iteración.

4.1.5 Organización del Proyecto

Participante en el proyecto.- Corresponde al desarrollador del presente proyecto.

Tutor.- Es la persona encargada de verificar la veracidad del sistema, además de colaborar en la construcción de la arquitectura del sistema.

Programador.- Los conocimientos que poseen y con el que debe cumplir este perfil es de: conocimiento amplio en manejo de herramientas de desarrollo para aplicativos web.

Administrador.- Es la persona encargada de gestionar los procesos como también de administrar la base de datos.

Usuarios Externos.- Son aquellas personas que utilizarán el sistema.

TABLA 4 PERFILES DE USUARIO DEL SISTEMA

PUESTO	RESPONSABILIDAD
Tutor	<p>Tiene una importancia vital ya que es él, quien asume la supervisión del proyecto encargándose así de proveer del personal necesario, los recursos imprescindibles y de tomar las decisiones que ayuden a que el proyecto cumpla con los objetivos propuestos. Es por esta razón que es muy importante que el tutor esté familiarizado con la gestión de proyectos y todas las técnicas y herramientas que la componen y que le facilitarán su labor al trabajar en un proyecto.</p>
Programador	<p>Encargada de desarrollar todo el sistema, modelo de datos, modelo del sistema, seguridad y accesibilidad.</p>
Administrador	<p>Gestiona y administra el sistema, además de planificar conforme a las necesidades un plan de respaldo periódico de la base de datos.</p>
Usuarios Externos	<p>Encargados del uso e iteración adecuada con el sistema.</p>

Roles y Responsabilidades.- En sus diferentes puestos, el cumplimiento de sus tareas de responsabilidad permitirán obtener el

resultado esperado, por lo que se detalla las responsabilidades correspondientes en el siguiente cuadro:

4.1.6 Plan del Proyecto

Se presenta la estructura en fases, iteraciones y el calendario del proyecto.

Plan de Fases.- El proceso se llevará a cabo en base a etapas con una o más iteraciones en cada una de ellas. La siguiente tabla muestra la distribución de tiempos y el número de iteraciones de cada fase.

TABLA 5 PLANIFICACIÓN POR FASES

Fase	Iteraciones	Duración
Fase Inicial	1	6 semanas
Fase de Elaboración	1	8 semanas
Fase de Construcción	5	40 semanas
Fase de Transición	1	2 semanas

Los límites que marcan el final de cada fase se detallan como sigue:

Fase Inicial.- Aquí se desarrollarán los requerimientos del modelo desde la perspectiva del usuario tanto para los requisitos como para los

resultados finales que se quieren conseguir, lo que permite definir algunas particularidades importantes frente a este aspecto:

- ✓ Diseño.
- ✓ Tiempo-costo.
- ✓ Satisfacción del usuario.
- ✓ Compatibilidad del modelo.
- ✓ Forma de acceso.
- ✓ Precisión y calidad.
- ✓ Características específicas de cada módulo.
- ✓ Usuarios y clientes.
- ✓ Recursos necesarios.

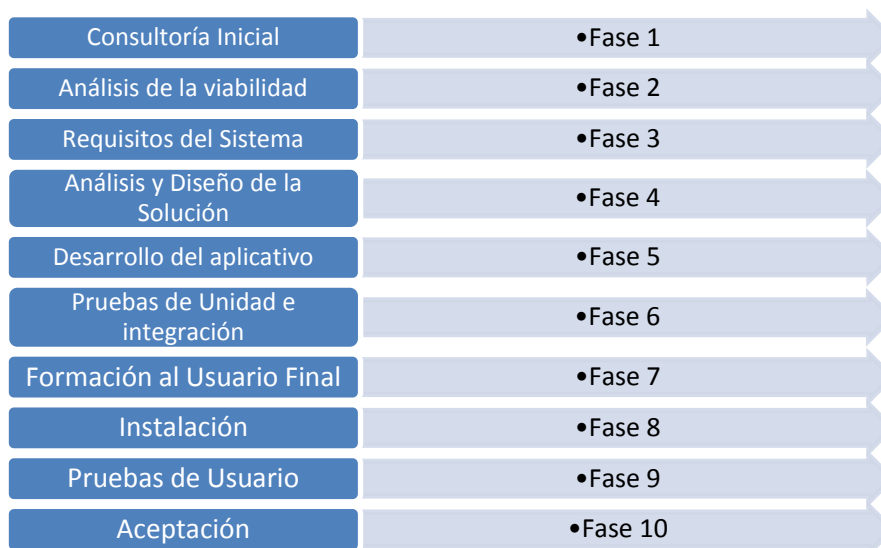
Fase de Elaboración.- Se consideran las necesidades y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y/o críticas del sistema). Como consecuencia de esta fase, tendremos bien identificados los casos de uso que define el sistema en su iteración con el usuario final.

Fase de Construcción.- Durante la fase de desarrollo se terminan de investigar y diseñar todos los casos de uso, refinando el modelo de análisis / diseño, el producto se construye en base a iteraciones validadas con el usuario final.

Fase de Transición.- En esta fase se presentará el entregable final para su distribución, asegurando un establecimiento y cambio del sistema previo de modo adecuado, incluyendo la instrucción de los usuarios. El límite que marca el fin de esta etapa, contiene la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, así como también se considera el cumplimiento de la instrucción de los usuarios.

Calendario del Proyecto. A continuación, se despliega un calendario de las primordiales tareas del proyecto incluyendo sólo las etapas de inicio y elaboración. Como se ha explicado, el proceso iterativo e incremental de RUP está determinado por la ejecución en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la mayoría de los artefactos son concebidos muy tempranamente en el proyecto pero van desarrollándose en mayor o menor escala de acuerdo a la fase e iteración del proyecto.

FIGURA 4.1 FASES DEL PROYECTO



Se ha determinado el siguiente calendario. La fecha de conformidad indica cuándo el artefacto en cuestión tiene un cambio de completitud suficiente para someterse a revisión y conformidad, pero esto no quita la posibilidad de su posterior refinamiento y cambios.

TABLA 6 CALENDARIO DEL PROYECTO

Disciplinas y Artefactos por Fases	Comienzo	Aprobación
Consultoría Inicial		
Recopilación de Información	Semana 3	Aprobado
Modelo de Casos de Uso	Semana 3	Aprobado
Análisis de Viabilidad		
Análisis y Selección de Plataforma	Semana 3	Aprobado
Requisitos del Sistema		
Modelos de Casos de Uso	Semana 5	semana 5
Especificaciones de Casos de Uso	Semana 6	Semana 6
Análisis y Diseño de Solución		
Modelo de Análisis y Diseño	Semana 7	Semana 9
Modelo de Datos	Semana 9	Semana 10
Desarrollo del Apicativo		
Desarrollo de aplicación	Semana 10	Semana 11
Prototipos de Interface de Usuario	Semana 11	Semana 13
Pruebas de Unidad e Integración		
Pruebas de Casos de Uso	Semana 13	Semana 14

Disciplinas y Artefactos por Fases	Comienzo	Aprobación
Formación al Usuario Final		
Capacitación	Semana 15	Semana 15
Instalación		
Implementación	Semana 15	Semana 15
Pruebas de Usuario		
Retroalimentación	Semana 16	Semana 16
Aceptación		
Entregables Finales	Semana 16	Semana 16

Seguimiento y control del proyecto.- El seguimiento del proyecto permitirá ajustar el desarrollo del aplicativo al calendario programado cumpliendo secuencialmente las fases definidas. Así deducimos que se deba cumplir con:

Gestión de requisitos.- Los requerimientos del sistema son especificados en el Capítulo III de este documento. Cada requerimiento tendrá una serie de particularidades tales como importancia, estado, iteración donde se implementa, etc.

Estas propiedades permitirán efectuar una búsqueda segura de cada requisito. Los cambios en los requerimientos serán gestionados mediante una solicitud de cambio, las cuales serán evaluadas y

distribuidas para asegurar la integridad del sistema y el correcto proceso de gestión de configuración y cambios.

Control de plazos.- El calendario del plan tiene un seguimiento semanal por el tutor encargado de revisión del proyecto.

Control de calidad.- Las desventajas detectadas en las revisiones y estipuladas también en una solicitud de cambio tendrán un seguimiento para asegurar la conformidad respecto de la solución de dichas deficiencias.

Gestión de riesgos.- A partir de la fase de inicio se conservará una lista de conflictos asociados al plan y de las acciones establecidas como estrategia para mitigarlos o acciones de contingencia.

4.2 ANÁLISIS Y ELABORACIÓN DEL SISTEMA

Esta fase permite definir el producto a desarrollar, especificando los procesos y estructuras de datos que se van a emplear.

Esto se logra con una adecuada comunicación entre el analista programador y el usuario final a fin de satisfacer todas las necesidades que pueda solventar el aplicativo en su futura implementación.

En el caso de falta de información por parte del usuario se puede recurrir al desarrollo de prototipos para saber con más precisión sus requerimientos. El análisis estructurado se pueden emplear varias técnicas como:

Diagramas de Flujo de datos.- Sirven para conocer el comportamiento del sistema mediante representaciones gráficas.

Modelos de datos.- Sirven para conocer las estructuras de datos y sus características. (Entidad relación y formas normales).

Diccionario de datos.- Sirven para describir todos los objetos utilizados en los gráficos, así como las estructuras de datos.

Definición de los interfaces de usuario.- Sirven para determinar la información de entrada y salida de datos.

Es importante un correcto análisis del aplicativo a desarrollar para descartar la posibilidad de omisión y/o errores que lleven a diseños no adecuados.

4.2.1 Definición del Sistema

Para realizar la definición de la aplicación fue esencial realizar trabajo de campo en donde se definieron los requerimientos del sistema DIRGES, lo que permitió que la aplicación se desarrolle

en función a las especificaciones observando las particularidades que la gestión tanto del recurso humano como de la distribución de insumos de bodega en el ámbito militar demanda.

4.2.2 Requisitos Básicos del Sistema

Requisitos legales.- En virtud que el sistema procesará datos personales del recurso humano asignado a la DIRGES, se deberá cumplir con todas las normativas vigentes en protección de datos y acceso a la información, así como también el sistema deberá garantizar la confidencialidad de la información, por lo tanto, el diseño deberá contemplar los métodos de identificación y control de acceso a la información de acuerdo a la ley y con los mecanismos de certificación, control/cifrado adecuados para cumplir lo establecido legalmente.

Requisitos de propiedad intelectual y Licencias.- El proyecto debe basarse en software libre utilizando herramientas OpenSource o código abierto. De ser necesario con licencias lo menos restrictivas posible, teniendo en cuenta las inversiones necesarias, tanto en la implementación inicial como en futuras actualizaciones que se requieran, cumpliendo así con las disposiciones gubernamentales para la implementación de software en las instituciones públicas.

Requisitos de acceso único.- La autenticación será mediante una única cuenta y una palabra clave para el acceso a los servicios desde el portal web de DIRGES.

Requisitos sobre la base de datos.- La base de datos de la empresa será de acceso exclusivo del personal técnico informático que administrará el aplicativo, ya que contendrá información sensible sobre existencias en bodega, y la información personal del recurso humano que labora en dichas dependencias. Esta base de datos podrá accederse a través de peticiones SQL desde el aplicativo web.

Requisitos del sistema de seguridad.- El sistema informático de la DIRGES (servidores y máquinas clientes) deberá tener un diseño basado en un cortafuego institucional, y con su correcto enrutamiento en los servidores proxy a fin de optimizar el acceso desde la intranet.

La red interna de la DIRGES posee una distribución de direcciones IP estáticas y un servidor de nombres secundario interno (DNS), lo que contribuye a la seguridad e identificación del punto remoto que interactúa con el aplicativo.

Requisitos tecnológicos, mantenimiento y administración.- Es importante para el buen funcionamiento posterior a la implementación del aplicativo, una correcta formación de los administradores y de los usuarios.

Esta formación podrá realizarse durante los períodos de prueba para generar confianza y para que acepten la nueva tecnología sin preconceptos ni falsas expectativas. Al utilizarse una plataforma web, el sistema no limita el número de usuarios para acceder al mismo, sin embargo el sistema inicialmente tendrá las siguientes necesidades en cuanto a equipos e infraestructura física.

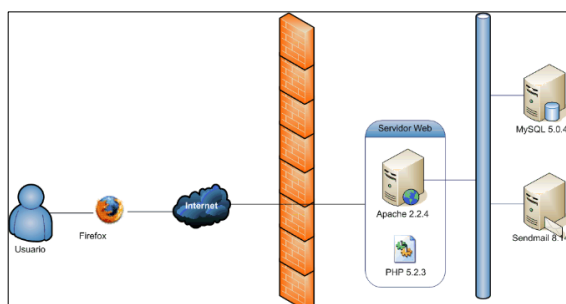


FIGURA 4.2 REQUISITOS TECNOLÓGICOS DEL SISTEMA

Requisitos de organización. La DIRGES elaborará un organigrama de recursos humanos para redefinir las tareas y las responsabilidades en el nuevo sistema. Esta reorganización tendrá por objetivo asignar las nuevas responsabilidades entre el personal naval de DIRGES.

Requisitos de seguridad. La empresa, además, deberá contar con un documento de seguridad que describa tanto las responsabilidades de los usuarios que trabajan con información sensible, como los procedimientos que hay que seguir en caso de que se descubran problemas relativos a la seguridad (intrusos, pérdida de confianza, vulnerabilidades, etc.) con el fin de incluir todos los pasos que deben seguir los responsables ante situaciones de este tipo.

4.2.3 Análisis y Casos de Uso

Caso de Uso: Control de Acceso al Sistema.

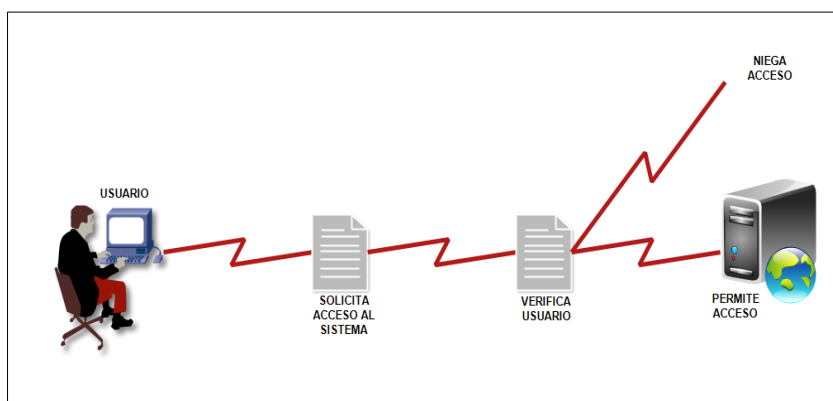


FIGURA 4.3 CONTROL DE ACCESO AL SISTEMA

Descripción breve.- Es el proceso en el cual el usuario ingresa sus datos y se valida dicha información para poder acceder a la aplicación.

Flujo básico de eventos.-

- El usuario debe ingresar con su nombre de usuario y clave para poder acceder al sistema.

- El sistema verificará la existencia del usuario en la base de datos.
- En caso de que los datos sean correctos la aplicación activará a las opciones que tiene permiso cada perfil de este usuario.
- Si los datos son incorrectos se mostrará un mensaje bloqueando el acceso al sistema.

Flujos alternativos.-

- El usuario abandona el sistema al cerrar sesión en cualquier momento.

Precondiciones.-

- Tener acceso al sistema.
- Tener creado usuario y estar activo en el sistema.

Post condiciones.-

- Ninguna.

Caso de Uso: Administración de Usuarios.

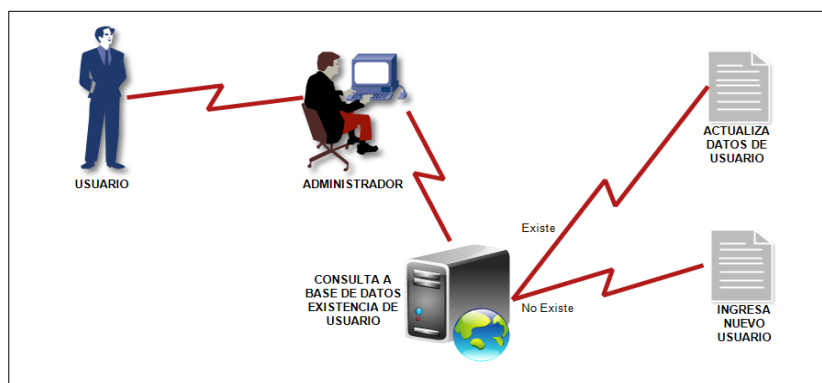


FIGURA 4.4 CASO DE USO: ADMINISTRACIÓN DE USUARIOS

Descripción breve.- Es el proceso en el cual el usuario bajo perfil de administrador ingresa nuevos usuarios al sistema o a su vez edita la información de los usuarios del sistema previamente grabados.

Flujo básico de eventos.-

- El usuario administrador visualiza la lista de usuarios existentes en el sistema mediante una lista tipo tabla.
- En caso de no existir el usuario requerido se procederá a ingresar un usuario nuevo mediante un formulario de entrada.
- En caso de requerirse actualizar las cuentas de usuario existentes, se deberá seleccionar de la lista tipo tabla, y los datos de mencionado usuario se cargarán en los controles editables de un formulario.
- Una vez ingresados o editados los datos de la cuenta de usuario requerida se procederá a grabar el registro en la base de datos.
- Si los datos son correctos, se visualizará un mensaje indicando que el registro se grabó correctamente.

Flujos alternativos.-

- El usuario puede cambiarse de página cerrando el módulo actual, pero la información no se guardará.

Precondiciones.-

- Tener acceso al sistema.

- Tener creado el usuario bajo perfil de administrador.

Post condiciones.-

- Ninguna.

Caso de Uso: Gestión de Licencias y Permisos.

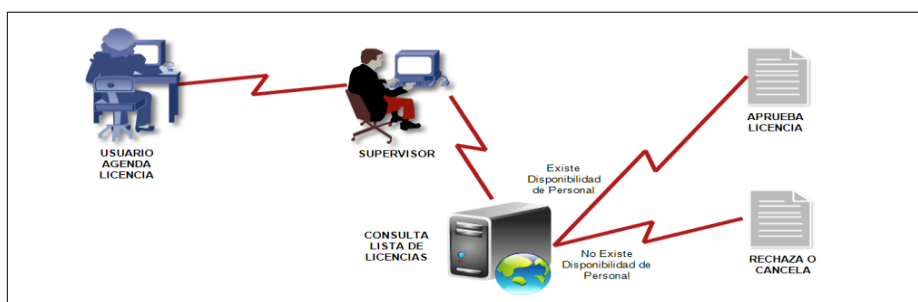


FIGURA 4.5 CASO DE USO: GESTIÓN DE LICENCIAS Y PERMISOS

Descripción breve.- Es el proceso con el cual el usuario supervisor de recursos humanos aprueba, rechaza o cancela las peticiones de licencias o permisos de los usuarios del sistema.

Flujo básico de eventos.-

- Mantener un flujo adecuado del personal con licencia y del personal que labora durante el año.
- Se verifica si el usuario aplica la licencia previamente programada.
- Se analiza el personal por departamentos, a fin de garantizar el equilibrio del personal en todas las subunidades de la institución.
- De existir disponibilidad de personal, se asigna dicho permiso, caso contrario se rechaza o se cancela adjuntando la debida

justificación de la decisión tomada para conocimiento del usuario interesado.

Flujos alternativos.-

- El usuario puede abandonar el presente módulo cuando lo requiera o cuando la sesión expira debido a la inactividad del mismo.

Precondiciones.-

- Tener acceso al sistema.
- Tener creado el usuario bajo perfil de administrador.
- Estar logueado en el sistema.

Post condiciones.-

- Ninguna.

Caso de Uso: Administración de Catálogo.-

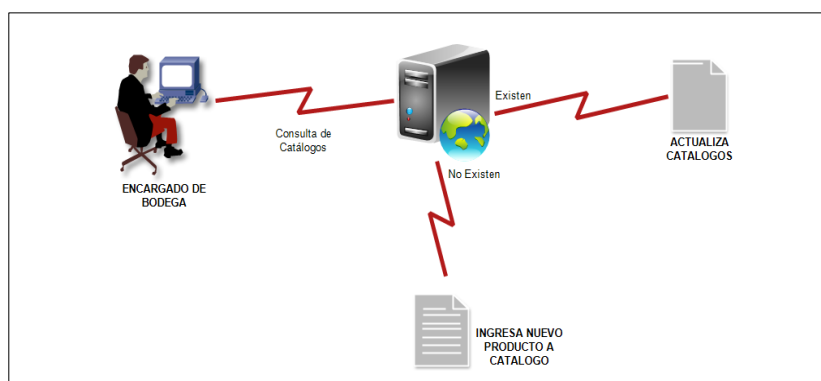


FIGURA 4.6 CASO DE USO: ADMINISTRACIÓN DE CATÁLOGO

Descripción breve.- Es el proceso con el cual se administra los productos que serán utilizados por la empresa como útiles de oficina, de aseo entre otros. La correcta administración de catálogo permitirá un flujo adecuado de mercadería a fin de evitar daños en la misma por caducidad o tiempo de vida útil.

Flujo básico de eventos.-

- Ingresar nuevos productos.
- Consultar mediante búsquedas inteligentes cualquier producto.
- Administrar unidades de medida de los productos.
- Administrar categorías de los productos.
- Administrar proveedores.
- Imprimir los productos existentes en bodega.

Flujos alternativos.-

- El usuario abandona el sistema al cerrar sesión en cualquier momento.

Precondiciones.-

- Tener acceso al sistema.
- Tener perfil de administrador y además estar activo.

Post condiciones.-

- Ninguna.

Caso de Uso: Entrada y Salida de Artículos de Bodega.

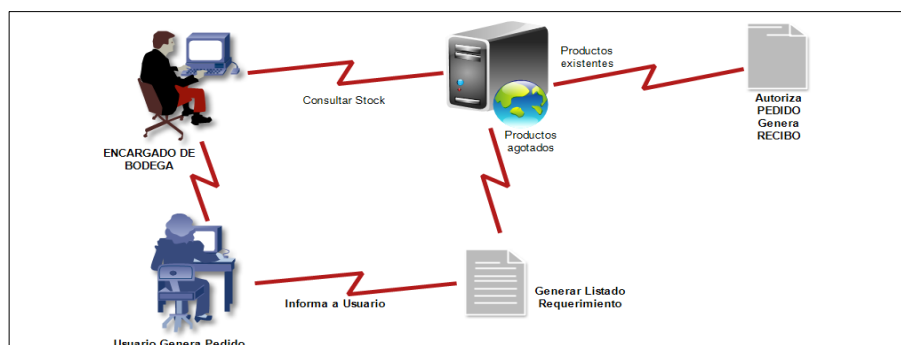


FIGURA 4.7 CASO DE USO: ENTRADA Y SALIDA DE ARTÍCULOS DE BODEGA

Descripción breve.- Es el proceso con el cual se administran las compras y salidas de cada uno de los productos existentes en bodega. Además es un registro de manera organizada de la mercadería que se tiene en una bodega.

Flujo básico de eventos.-

- Realizar compras de productos y administrar el stock del producto.
- Consultar el detalle del Kardex de cada producto.
- Administrar pedidos de los productos por parte de los empleados de la empresa.
- Imprimir el Kardex del producto.

Flujos alternativos.-

- El usuario puede abandonar el presente Submódulo cuando lo requiera o cuando la sesión expira debido a la inactividad del mismo.

Precondiciones.-

- Tener acceso al sistema.
- Tener creado el usuario bajo perfil de administrador.
- Estar logueado en el sistema.

Post condiciones.-

- Ninguna.

Caso de Uso: Cuentas por Pagar.

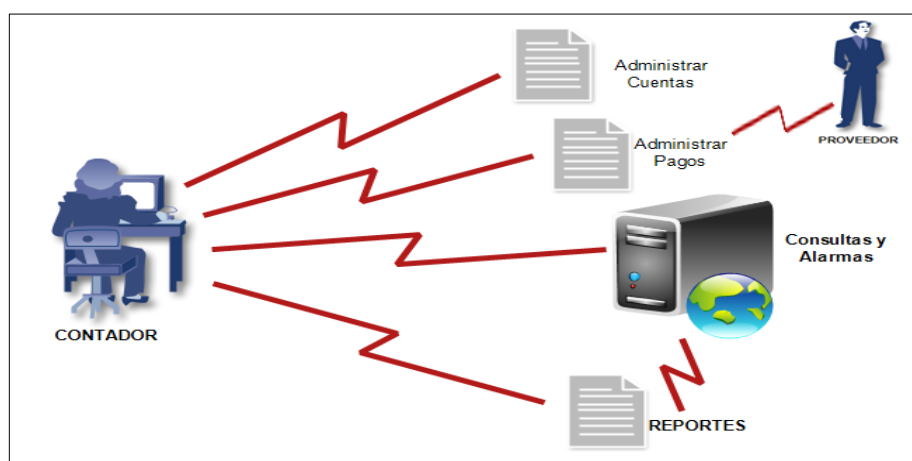


FIGURA 2 CASO DE USO: CUENTAS POR PAGAR

Descripción breve.- Es el proceso con el cual se administran los créditos y deudas que tiene la empresa a los proveedores, en este caso especial los proveedores de productos de oficina, suministros de computación, aseo y limpieza, lubricantes, etc.

Flujo básico de eventos.-

- Administrar créditos.

- Administrar pagos, este proceso permitirá realizar el pago de acuerdo al crédito que tiene la empresa con los proveedores.
- Consultar mediante búsquedas inteligentes el estado de los créditos.
- Se emitirá alarmas mensuales de las deudas vencidas de pago.

Flujos alternativos.-

- El usuario puede abandonar el presente módulo cuando lo requiera o cuando la sesión expira debido a la inactividad del mismo.

Precondiciones.-

- Tener acceso al sistema.
- Tener creado el usuario bajo perfil de administrador.
- Estar logueado en el sistema.

Post condiciones.-

- Ninguna.

4.3 DISEÑO LÓGICO DEL SISTEMA

El diseño lógico de sistemas se refiere a lo que hará el nuevo sistema, es una descripción de los requerimientos funcionales de un sistema. Es decir,

es la expresión conceptual de lo que hará el sistema para solucionar las dificultades identificadas en el análisis previo.

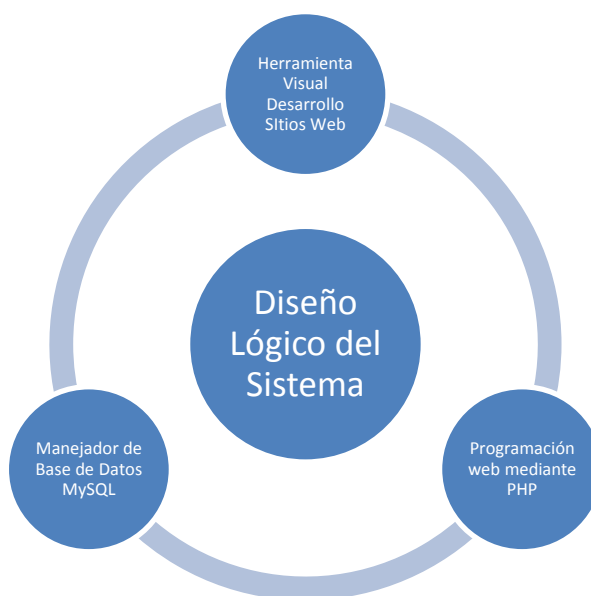


FIGURA 4.9 DISEÑO LÓGICO DEL SISTEMA

El objetivo de la fase de diseño de un proyecto de estas características es obtener los componentes definidos en la etapa de análisis. Básicamente, se debe tener en cuenta que en esta etapa se inicia el proceso de selección de los servicios y su distribución física, y se analiza cómo serán ejecutados estos servicios en una arquitectura hardware. Además, se deberán determinar las especificaciones de desarrollo e integración, así como definir el entorno de pruebas y seleccionar qué criterios se utilizarán para que éstas sean representativas del correcto funcionamiento del sistema.

El diseño lógico contiene proyectar el propósito de cada elemento del sistema, sin relación con consideraciones de hardware y software. Las especificaciones de diseño lógico que se determinan, y documentan se definen en los párrafos siguientes:

Diseño de salida.- Es una descripción de todas las salidas del sistema y contiene sus tipos, formato, contenido y frecuencia. Por ejemplo, la necesidad de que todas las facturas de la compañía contengan el número de factura original de los clientes es una especificación de diseño lógico. Los instrumentos de diseño de pantallas e informes pueden utilizarse durante la fase de diseño de salidas para satisfacer los requerimientos de salidas del sistema.

Diseño de entrada.- Una vez que se completa el diseño de salidas, puede iniciarse el de entradas. Aquí se especifican los tipos, formato, contenido sistema capture los números telefónicos de los clientes cuando éstos llaman a la organización y use tal dato para encontrar de manera automática la información de su cuenta, es una especificación de diseño lógico.

Es viable utilizar diagramas y diseños de pantallas e informes diversos para detallar el tipo, formato y contenido de los datos de entrada.

Diseño de procesamiento.- Los tipos de cálculos, comparaciones y manipulaciones de datos en general que demande el sistema se establecen

durante esta etapa. Por ejemplo, un programa de nómina requiere cálculos de los sueldos brutos y netos, retenciones de impuestos federales y locales, y otras deducciones y planes de ahorro.

Diseño de archivos y bases de datos.- En diversos sistemas de información se requieren subsistemas de archivos y bases de datos. Las características de estos subsistemas se definen también en la fase de diseño lógico. Por ejemplo, la capacidad para lograr la actualización inmediata de los registros de los clientes es una especificación de diseño lógico. En muchos casos, un administrador de bases de datos opina en este aspecto del diseño. Los diagramas de flujo de datos y de entidad relación por lo común se utilizan durante el diseño de archivos y bases de datos.

Diseño de telecomunicaciones.- Durante el diseño lógico es preciso especificar los sistemas de redes y telecomunicaciones. Por ejemplo, en un hotel podría detallarse un sistema de clientes servidor con un cierto número de estaciones de trabajo vinculadas con el servidor. A partir de estos requerimientos, podría elegirse por una topología híbrida. Los programas de gráficos y las herramientas de CASE son útiles para proveer el diseño de redes lógicas.

Diseño de procedimientos.- Todo sistema de información demanda procedimientos para la realización de aplicaciones y la solución de los inconvenientes que surjan. Estos requerimientos significativos se capturan

durante el diseño de procedimientos. Una vez delineados, los procedimientos se pueden describir con programas de procesamiento de texto. A manera de ejemplo, los pasos necesarios para aumentar una nueva cuenta de cliente alcanzarían incluir una serie de trabajos manuales y computarizados. Deben redactarse procedimientos escritos para que sean eficaces y todo mundo los siga.

Diseño de controles y seguridad.- Otra parte importante del diseño lógico es establecer la frecuencia y características necesarias de los sistemas de respaldo. En general, debe tenerse soporte de todo, lo que incluye el hardware, software, datos, personal, insumos e instalaciones.

Además, en esta fase del diseño lógico ha de considerarse la planeación de cómo advertir un desastre del equipo de cómputo y la forma de recuperarse de él si ocurre.

Diseño de personal y empleos.- Algunos sistemas requieren contratar empleados adicionales, mientras que con otros es preciso cambiar las tareas relacionadas con uno o más empleos de SI existentes.

Los nombres y descripciones de los puestos se detallan durante el diseño de personal y empleos. Los organigramas son ventajosos en el diseño de personal para diagramar los empleos y sus nombres. También se utilizan procesadores de textos para detallar las funciones de cada puesto.

Se pueden resumir las tareas de esta fase como:

- ✓ Arquitectura del sistema: componentes de hardware, su interconexión, jerarquía software, seguridad y privilegios, todo lo que es necesario para que el sistema pueda ser configurado y puesto en marcha.
- ✓ Estructuración y Diseño de la base de datos.

4.3.1 Arquitectura

El sistema se basa en una arquitectura de tres capas. La **programación por capas** es una arquitectura cliente-servidor en el que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario. [16]

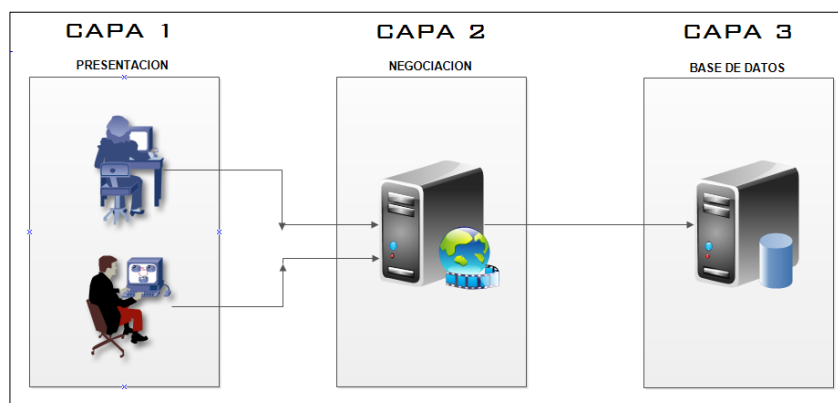


FIGURA 4.10 ARQUITECTURA CLIENTE – SERVIDOR

La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, solo se ataca al nivel requerido sin tener que revisar entre el código

mezclado. En el diseño de sistemas informáticos actual se suelen usar las arquitecturas multinivel o programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

- ✓ **Capa 1.-** Corresponde al diseño del sistema, todos los archivos del tema en la carpeta "themes". En esta capa se consideran los usuarios finales en los diferentes puntos de acceso al sistema mediante Internet.

- ✓ **Capa 2.-** Corresponde al Control de Datos, el contenido proporcionado por el usuario es controlado por los archivos en la carpeta raíz. El acceso a la capa 2 puede ser local o vía internet mediante una cuenta de administrador del sistema.

- ✓ **Capa 3.-** Es donde residen los datos y es la encargada de acceder a los mismos. El acceso a la base de datos es controlado mediante archivos en la carpeta "classes".

La capa 3 corresponde al servidor central del sistema, alojado en las instalaciones de la DIRGES desde donde se centraliza el servicio para todos los retenes navales de Esmeraldas.

4.3.2 Modelo Físico de Datos

El sistema maneja una base de datos para el módulo de recursos humanos y otra base de datos para el módulo de gestión de bodega.

Diagrama Entidad – Relación.-

Para un mejor entendimiento se ha dividido en partes:

Website.-

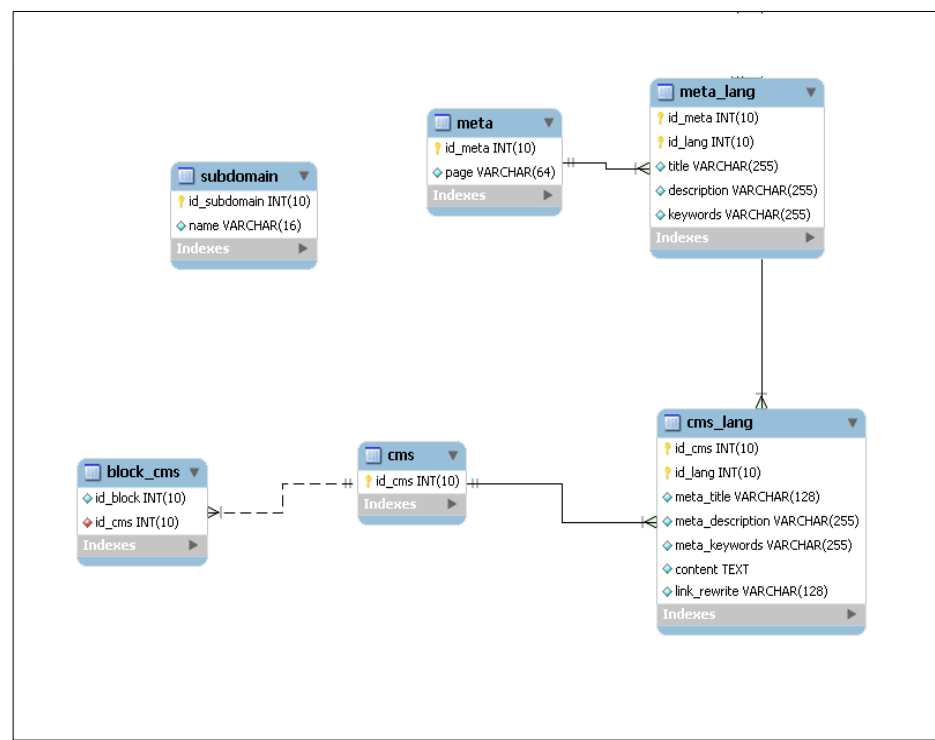


FIGURA 4.11 TABLAS CON DATOS DE WEBSITE

Módulos.-

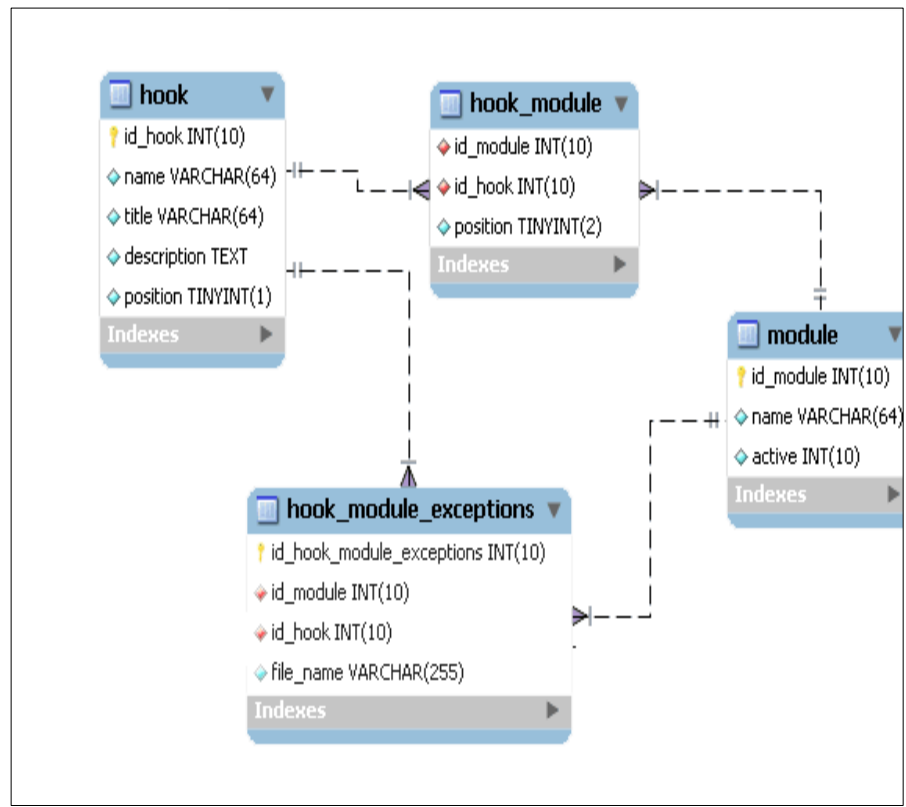


FIGURA 4.12 TABLAS CON DATOS DE MÓDULOS

Cientes.-

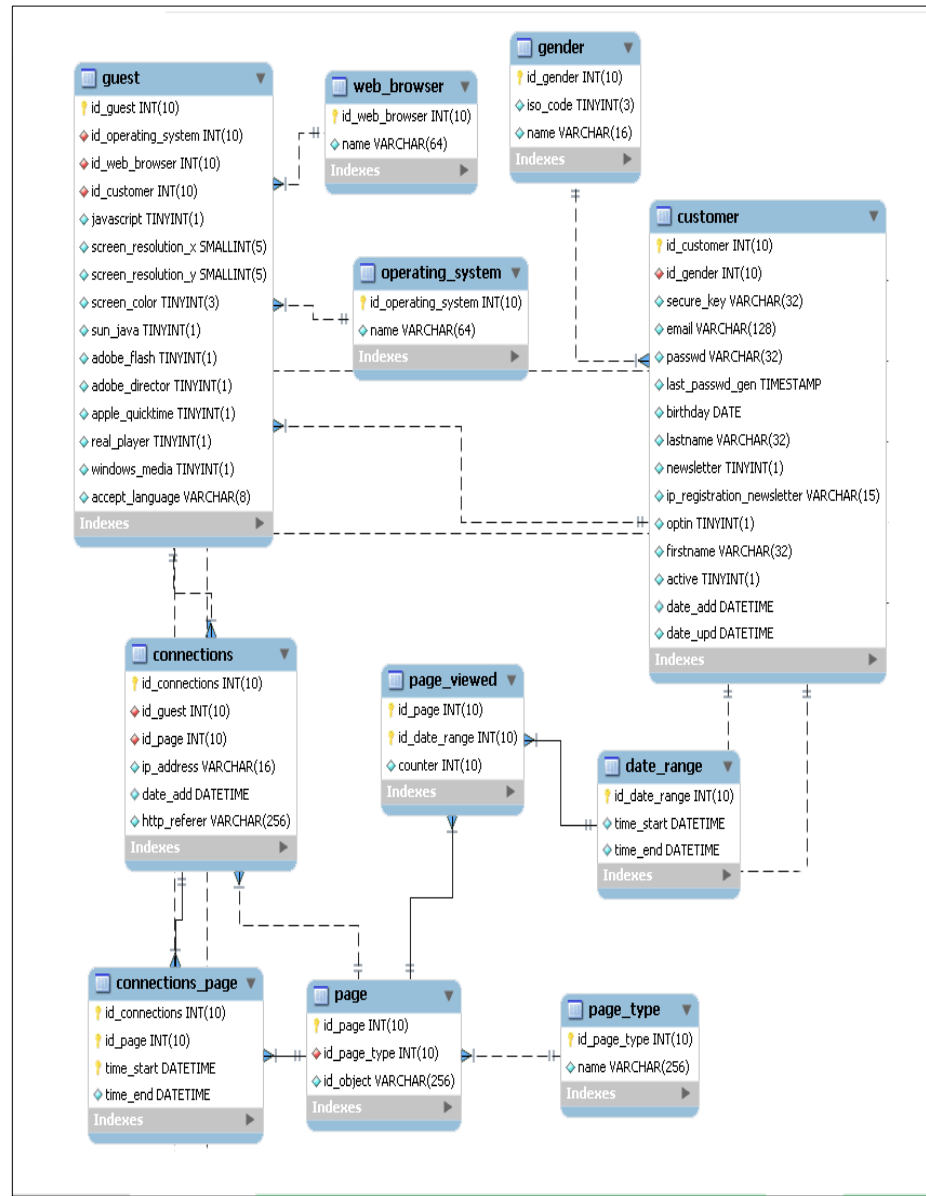


FIGURA 4.13 TABLAS CON DATOS DE CLIENTES

Productos.-

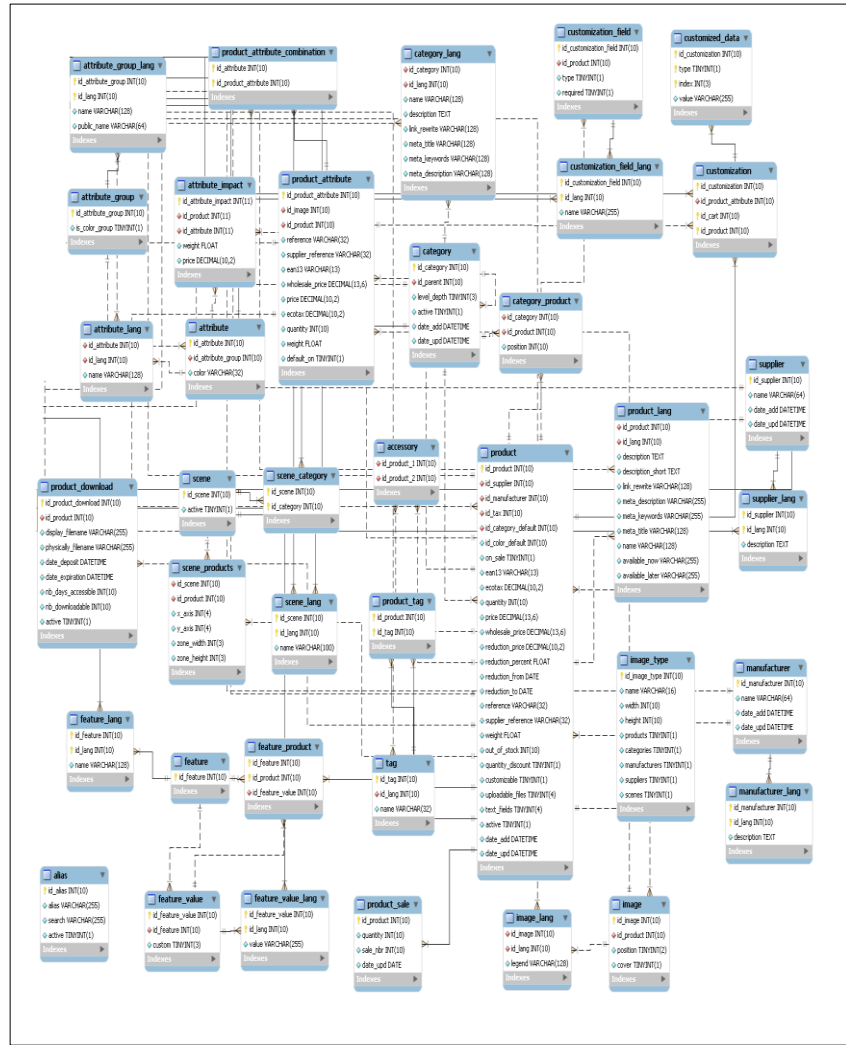


FIGURA 4.14 TABLAS CON DATOS DE PRODUCTOS

4.4 DESARROLLO DEL SISTEMA

4.4.1 Planificación de las Actividades de Desarrollo

Al momento de realizar la planificación se ha tenido en cuenta los siguientes puntos:

Análisis del Panorama. Donde realizamos la descripción general del proyecto, detalle de la organización del plan y resumen del resto de la documentación.

Análisis de Fases. Se analiza el ciclo de desarrollo del proyecto como es: análisis de requisitos, fase de diseño de alto nivel, fase de diseño de bajo nivel, etc.

Asociada con cada fase debe de haber una fecha que especifique cuando se debe terminar estas fases y una indicación de cómo se pueden solapar las distintas fases del proyecto.

Análisis de organización.- Se definen las responsabilidades específicas de los grupos que intervienen en el proyecto.

Análisis de pruebas.- Se hace un esbozo general de las pruebas y de las herramientas, procedimientos y responsabilidades para realizar las pruebas del sistema.

Análisis de control de modificaciones.- Se establece un mecanismo para aplicar las modificaciones que se requieran a medida que se desarrolle el sistema.

Análisis de documentación.- Su función es definir y controlar la documentación asociada con el proyecto.

Análisis de capacitación.- Se describe la preparación por parte del desarrollador que participa en el proyecto y las instrucciones a los usuarios para la utilización del sistema que se les entregue.

Análisis de revisión e informes.- Se analiza cómo se informa del estado del proyecto y se definen las revisiones formales asociadas con el avance de proyecto.

Análisis de instalación y operación.- Se describe el procedimiento para instalar el sistema en la localidad de la empresa de rastro.

Análisis de mantenimiento.- Se establece un bosquejo de los posibles tipos de mantenimiento que se tienen que dar para futuras versiones del sistema.

4.4.2 Desarrollo y codificación

Para el diseño de una interfaz de usuario en un sistema Web, es necesario analizar los enlaces que debe tener la aplicación, pensando en las características principales que tendrá el sistema y luego de esto ir construyendo tantos enlaces a otras páginas Web como sea necesario. Sin embargo al ser una aplicación web se facilita el proceso de diseño de la aplicación ya que existen librerías que ayudan para una mejor presentación.

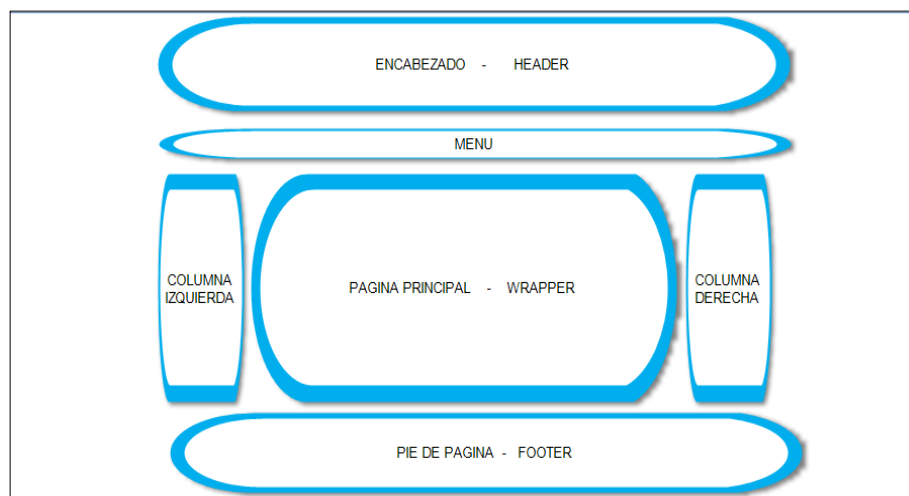


FIGURA 4.17 ESQUEMA DEL SITIO WEB

Encabezado.- En esta parte de la estructura de la aplicación va el logo del sistema, junto con el nombre del sistema.

Menú.- En esta ubicación se pondrá los menús de nuestra aplicación, los cuales nos permitirán navegar por la aplicación de una manera rápida y eficiente.

Cuerpo de la Aplicación.- Es esta parte de la estructura se realizarán todas las operaciones de la aplicación, es decir: Crear, modificar, imprimir, eliminar entre otras.

Footer.- En este lugar se desplegará los derechos de Autor como también la licencia del sistema. El sistema se basa en un motor de plantillas, el cual permite crear fácilmente un propio tema personalizado, mediante un conjunto de archivos que pueden editar con el fin de cambiar el aspecto de su sitio virtual.

Todos los temas tienen sus archivos ubicados en la carpeta raíz /themes. Cada tema contiene su propia sub-carpeta, en la carpeta de temas principales. Cada tema está compuesto de archivos plantilla (.tpl), archivos de imagen (.gif, .jpg, .png), uno o más archivos CSS (.css) e incluso archivos JavaScript (.js).

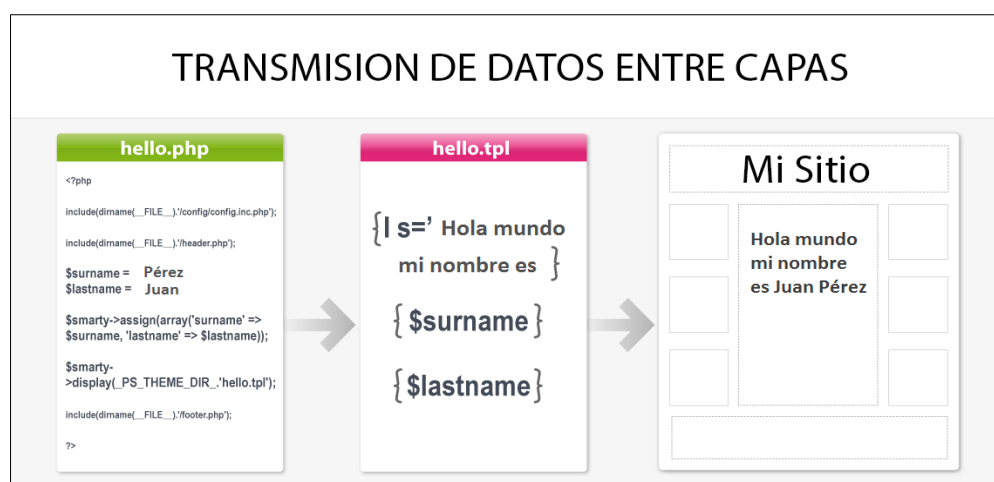


FIGURA 4.18 TRANSMISIÓN DE DATOS ENTRE CAPAS

Cada tema tiene su propio archivo de imagen preview.jpg en su carpeta, permitiendo observar la manera como luce el tema directamente, y seleccionar el tema apropiado.

Así tenemos un esquema de la estructura de archivos que conforman la interfaz de usuario del sistema:

- ✓ La carpeta /css contiene todos los archivos CSS.
- ✓ La carpeta /img contiene todas las imágenes.
- ✓ La carpeta /js contiene todos los archivos JavaScript.

- ✓ La carpeta /lang contiene traducciones del tema.
- ✓ La raíz de la carpeta sólo contiene archivos TPL así como el archivo preview.jpg.

Los datos de la conexión a la Base de Datos se especifican en el fichero **/config/settings.inc.php**, pudiendo también configurarse en la pestaña de tu sitio en la opción **Preferencias/Bases de datos**.

```

1 <?php
2 define('_DB_SERVER_', 'localhost');
3 define('_DB_NAME_', 'bodega');
4 define('_DB_USER_', 'dnunez');
5 define('_DB_PASSWD_', 'dnunez');
6 define('_DB_PREFIX_', 'ps ');
7 define('_MYSQL_ENGINE_', 'InnoDB');
8 define('_PS_CACHING_SYSTEM_', 'CacheMemcache');
9 define('_PS_CACHE_ENABLED_', '0');
10 define('_COOKIE_KEY_', 'OHu0TzMuggp0nkMqNqWhJoVYhvXe13bQApkuNOKf5p7rF3E0RsUGcHhN');
11 define('_COOKIE_IV_', 'BzKQcF2E');
12 define('_PS_CREATION_DATE_', '2015-03-03');
13 if (!defined('_PS_VERSION_'))
14     define('_PS_VERSION_', '1.6.0.13');
15 define('_RIJNDAEL_KEY_', 'CagqxxggC2XMr8bqd02Zg0fDAImhVwxkW');
16 define('_RIJNDAEL_IV_', 'PQoYIoMDHJz4OeS5pdU4Xw==');
17 |

```

riables indispensables a configurar serán:

- ✓ **_DB_SERVER_**: define el nombre del servidor.
- ✓ **_DB_NAME_**: define el nombre de la base de datos.
- ✓ **_DB_USER_**: define el usuario para el sistema.
- ✓ **_DB_PASSWD_**: define la clave del usuario del sistema.
- ✓ **_DB_PREFIX_**: define el prefijo que utilizaran las tablas de la base de datos.

CAPÍTULO 5

IMPLEMENTACIÓN Y PRUEBAS DEL SISTEMA

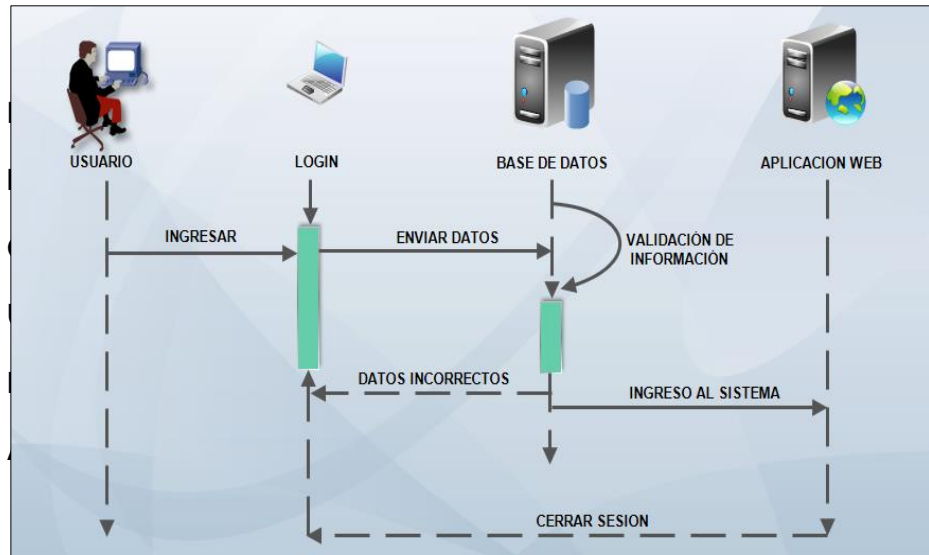
5.1 IMPLEMENTACIÓN

Culminado el desarrollo de la aplicación se procedió a implementar en un servidor de aplicaciones web de la DIRGES. El sistema consiste en una aplicación web modular, accesible a través de la Intranet de la DIRGES para todos los usuarios legalmente autorizados.

Cabe señalar que el sistema fue concebido para solventar necesidades de los retenes navales acantonados en la provincia de Esmeraldas, por lo que se habilita el acceso por internet desde cualquier punto remoto que lo requiera. La aplicación ha sido personalizada de acuerdo a las necesidades de la institución naval con sus particularidades.

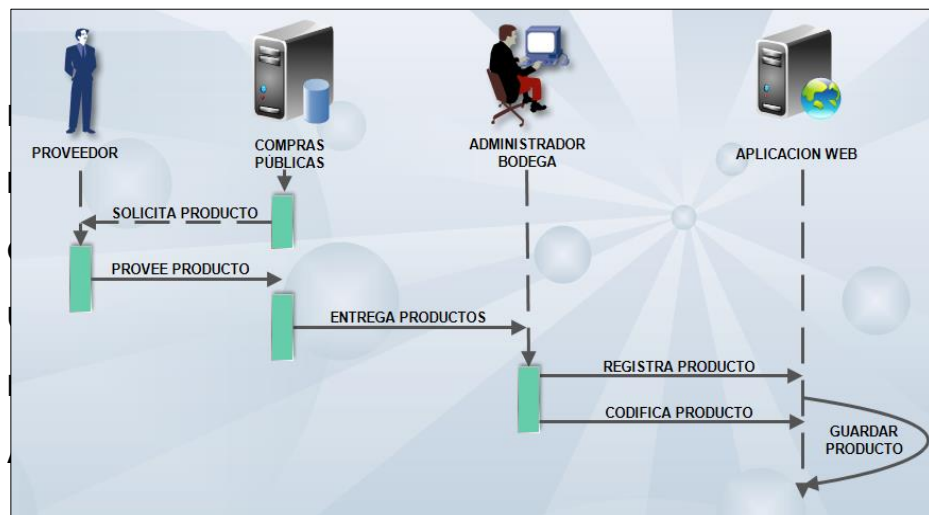
5.1.1 Diagrama de Secuencias

Diagrama de Secuencia.- Acceso al sistema



5.1 DIAGRAMA DE SECUENCIA: ACCESO AL SISTEMA

Diagrama de Secuencia.- Compra a Proveedores



**5.2 DIAGRAMA DE SECUENCIA: COMPRA A
PROVEEDORES.**

Diagrama de Secuencia.- Entrega de Productos

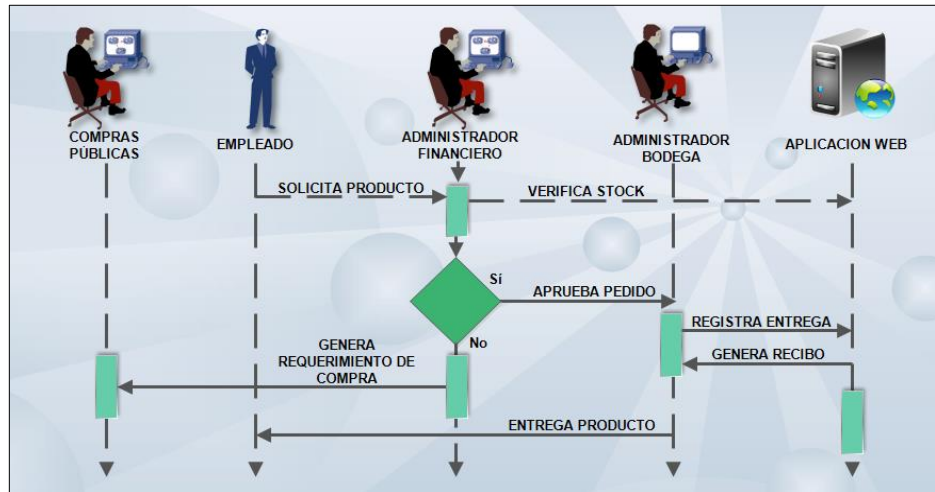


FIGURA 5.3 DIAGRAMA DE SECUENCIA: ENTREGA DE PRODUCTOS

Diagrama de Secuencia.- Gestión Aprobación Licencias y Permisos

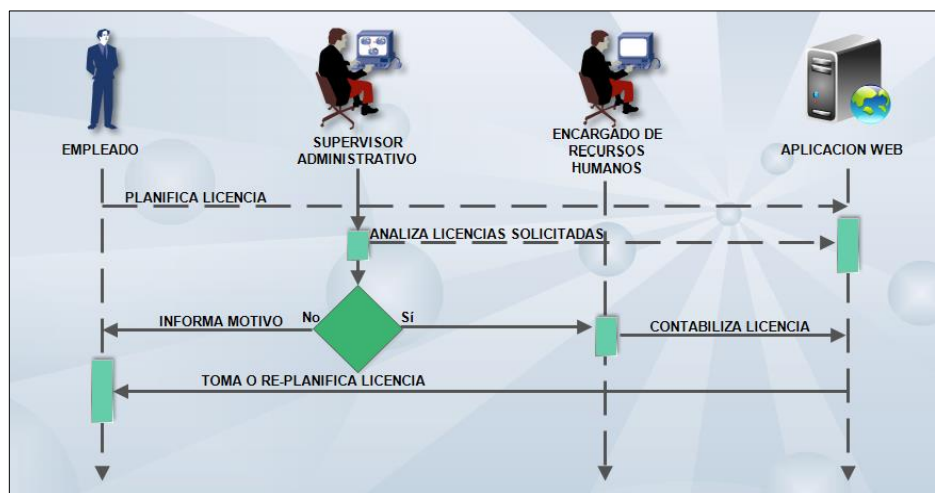


FIGURA 5.4 DIAGRAMA DE SECUENCIA: GESTION Y APROBACIÓN DE LICENCIAS

5.1.2 Diagrama de Actividades

Diagrama de Actividad.- Acceso al sistema.

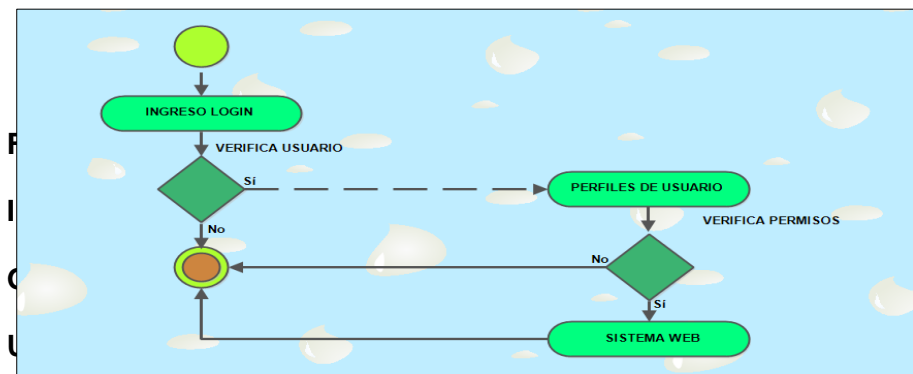


FIGURA 5.5 DIAGRAMA DE ACTIVIDADES: ACCESO AL SISTEMA

Diagrama de Actividad.- Pedidos de Productos.

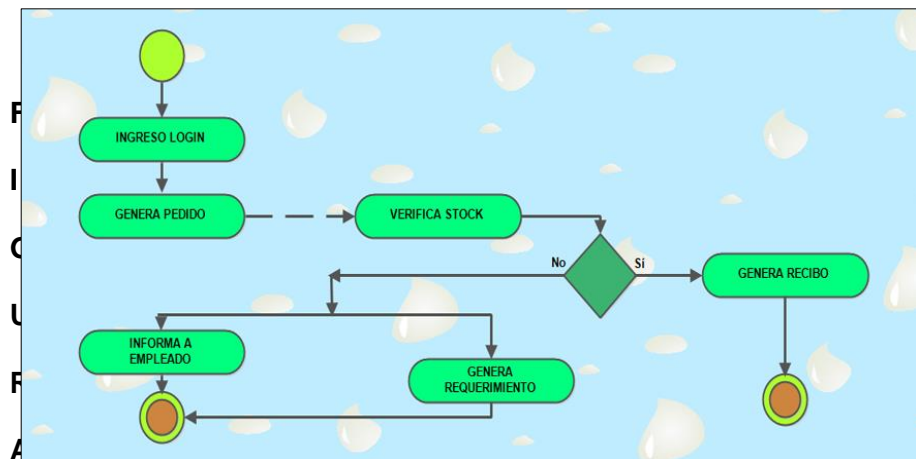


FIGURA 5.6 DIAGRAMA DE ACTIVIDADES: PEDIDO DE PRODUCTOS

Diagrama de Actividad.- Asignación de Licencias.

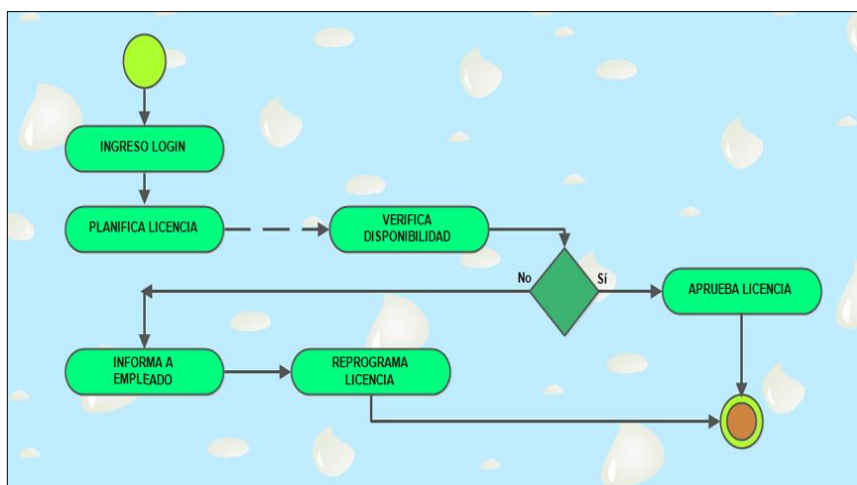


FIGURA 5.7 DIAGRAMA DE ACTIVIDADES: ASIGNACIÓN DE LICENCIAS

5.2 PRUEBAS

5.2.1 Caso de Prueba: Seguridad de la Aplicación

Descripción.- Para iniciar la aplicación se abrirá un navegador web, accediendo a la página de autenticación del usuario mediante una cuenta de usuario previamente creado en el sistema por el usuario administrador. Lógicamente todas las páginas del sistema tienen su respectiva ruta o url específica, por lo que el sistema permite controlar que el acceso a las mismas sea en forma secuencial evitando el ingreso específico a una determinada página sin observar el debido

orden de diseño del sistema, debido a que se empaqueta la aplicación en su correspondiente orden de ingreso.

Se debe tomar en cuenta que todos los usuarios pueden acceder al módulo de recursos humanos, no así al módulo de gestión de insumos, por lo que los encargados de retirar materiales del pañol de la DIRGES será el empleado responsable de la bodega de cada retén para efectos de legalizar sus egresos en el sistema, por lo que los usuarios del módulo de gestión de insumos pudieran ser diferentes a los usuarios del módulo de gestión de recursos humanos.

Comprobación.- Al momento de ingresar con un usuario que no exista en la base de datos del sistema, el usuario podrá apreciar una advertencia que le indica su intento de ingresar al sistema de forma fraudulenta. Del mismo modo, se puede copiar una url específica y pegarlo en la barra de direcciones de un navegador web, no pudiendo visualizarse la página deseada debido a que su ingreso no es el adecuado por su página antecesora.

Condiciones de ejecución.- Se puede verificar en el momento que el usuario estime conveniente.

Entrada.- Deberemos ingresar la dirección web de la aplicación:
<http://icaro.dirnea.org/dirges>.

- Ingresamos a la página de Servicios haciendo clic en el menú superior, o digitando directamente la dirección:
<http://localhost/dirges/services.html>.
- En cualquiera de las dos opciones: Gestión de Recursos Humanos o Gestión de Bodega, ingresamos datos de un usuario que no existe en la base de datos.

Resultado esperado.-

- No permite ingresar a usuarios inexistentes en la base de datos, sea el caso para el módulo de recursos humanos como para el módulo de gestión de insumos.
- Al tratar de ingresar directamente a una página del sistema sin estar autenticado, no permite ingresar y le expulsa de la aplicación.

Evaluación de prueba.- Prueba satisfactoria.

5.2.2 Caso de Prueba: Registro Egresos de Materiales

Descripción.- Este proceso se utilizará para entregar productos a los empleados que previamente hayan generado en el sistema el pedido de productos y que el administrador del mismo haya aprobado la entrega de dicho listado de materiales. Para realizar una entrega de productos, el usuario que necesita el producto deberá acercarse a la

bodega de la DIRGES con el ticket generado por el sistema en el que consta la lista de pedidos realizados, una vez que mencionada lista de productos ya ha sido verificada su existencia en stock, el bodeguero imprime un recibo de Entrega – Recepción de productos generado por el sistema para su posterior legalización y archivo, y finalmente despacha mencionado pedido y registra en el sistema.

Comprobación.- En esta prueba se despachó una lista de productos de limpieza y desinfectantes. El sistema emitió un número de pedido y posterior al despacho de mencionado pedido el sistema generó el recibo de egreso mismo que pudo ser impreso, legalizado por el usuario beneficiado, bodeguero y visto bueno del jefe financiero institucional.

Condiciones de ejecución.-

- El usuario deberá estar validado en el sistema y tener permiso para el módulo de Inventarios.
- Debe existir el producto.
- El producto debe tener stock para poder entregar, caso contrario deberá salir un mensaje de error indicando al usuario que no existe stock para este producto.

Entrada.- La entrega de productos se realiza en el menú PEDIDOS, bajo el perfil de administrador del sistema. Se interactúa con la siguiente información:

- Proveedores.
- Productos.
- Inventarios.

Resultado esperado.-

- Registro normal de las entregas de productos realizadas.
- Mensaje de error, en caso de no cumplir con los requisitos de este módulo.

Evaluación de prueba.-

- Prueba satisfactoria.

5.2.3 Caso de Prueba: Registro Ingreso de Materiales

Descripción.- El proceso que se realiza para ingresar productos o agregar unidades al stock, inicia desde la verificación en la base de datos si existe el producto o no por parte del usuario administrador de bodega, en caso de existir se registra el stock del producto; en caso de no existir el producto se ingresa un nuevo registro en el sistema con los datos de la compra: proveedor, precio, fecha de compra.

Comprobación.- Se probó el ingreso de 78 productos existentes en bodega. Cada ingreso se respalda con el acta de entrega – recepción y copia de la factura entregada por el Departamento Financiero a través de la División de Compras Públicas a fin de llevar un control adecuado de las adquisiciones de los mismos.

Condiciones de ejecución.- El ingreso de productos se puede verificar con un usuario administrador del sistema debidamente validado como tal en el mismo.

Entrada.- Deberemos ingresar la dirección web de la aplicación: <http://icaro.dirnea.org/dirges>, ingresando al sistema como usuario administrador.

- Ingresamos al menú CATALOGO.
- Verificamos la Lista de proveedores registrados en el menú PROVEEDORES.
- Verificación individual de stock por productos de acuerdo a las cantidades ingresadas.

Resultado esperado.-

- Se realizó un registro normal de los productos adquiridos.
- Se obtuvo los correspondientes mensajes de error al momento de ingresar equivocadamente los datos de los productos en los campos asignados para el efecto.

Evaluación de prueba.-

- Prueba satisfactoria.

5.2.4 Caso de Prueba: Reporte de Inventario de Bodega

Descripción.- Este proceso se utilizará para realizar cortes periódicos en el sistema a fin de verificar el stock de productos acorde a los documentos de descargo legalizados durante el mes. El encargado de bodega realizará la verificación de existencias en el sistema imprimiendo los reportes de stock de productos por categorías, para su posterior verificación con los documentos físicos archivados.

Comprobación.- Se contabilizó los egresos realizados en el primer mes de implementado el sistema, y posteriormente se verificó con los reportes generados por el sistema. El sistema generó las alarmas en los stocks cuyas cantidades eran las mínimas configuradas para mencionados productos.

Condiciones de ejecución.-

- El usuario deberá estar validado en el sistema y tener permiso para el módulo de Inventarios.
- Debe existir el producto.

Entrada.- La verificación de inventarios se realiza en el menú INVENTARIOS, bajo el perfil de administrador del sistema. Se interactúa con la siguiente información:

- Pedidos.
- Productos.
- Inventarios.

Resultado esperado.-

- Verificación correcta entre la información del sistema y los documentos de archivo.
- Mensajes de alerta, en caso de stock mínimo configurado.

Evaluación de prueba.-

- Prueba satisfactoria.

5.2.5 Caso de Prueba: Reporte de Permisos y Licencias

Descripción.- El proceso que se realiza para obtener los reportes de Permisos y Licencias desde el sistema inicia con la planificación de licencias por parte de los empleados. Dicha licencia puede ser aprobada, cancelada o rechazada por el administrador de recursos humanos o usuario supervisor del sistema, el mismo que podrá obtener el reporte consolidado de las peticiones de licencias generados, aceptados o rechazados. Se podrá aplicar filtro para los

reportes por subunidades o departamentos a fin de cumplir con el número mínimo de personal requerido en cada área de la institución.

Comprobación.- Se agenda 15 días de licencia para un empleado ABC, y posteriormente se aprobó mencionada licencia como usuario supervisor. Se agenda 7 días de permiso para un empleado 123 y mencionada licencia se rechazó por parte del usuario supervisor.

Condiciones de ejecución.- La planificación de las licencias se los hacen mediante un usuario normal, mientras que la aprobación o rechazo de las mismas se las hace con un usuario supervisor en el menú ASIGNAR LICENCIAS.

Entrada.- Deberemos ingresar la dirección web de la aplicación: <http://icaro.dirnea.org/dirges>, ingresando al sistema como usuario ABC y usuario 123.

- Agenda la licencia requerida en la opción LICENCIAS.
- Posterior, se ingresa al sistema como un usuario supervisor.
- .Aprobar o rechazar la licencia programada en la opción ASIGNAR LICENCIAS.

Resultado esperado.-

- Se realizó un registro normal de las licencias aprobadas y rechazadas realizadas por los usuarios ACB y 123.

- Se restó correctamente los 15 días de licencia aprobadas de los 30 días de licencia asignados al usuario ABC.
- Se notificó mediante un mensaje el motivo del rechazo a la petición de licencia por parte del usuario 123. Mencionados días no afectó a los 30 días asignados al usuario 123.

Evaluación de prueba.-

- Prueba satisfactoria.

CAPÍTULO 6

ANÁLISIS DE RESULTADOS

6.1 ANÁLISIS DE INFORMACIÓN RECOPIADA

Un análisis de impacto de resultados permite visualizar los resultados positivos mediante la medición de variables definidas antes y después de la implementación del sistema.

Para una mejor apreciación de mencionados resultados se analizará el impacto obtenido en cada una de las actividades que el sistema procesa.

6.1.1 Reportes de Bodega.-

Anteriormente se llevaba los registros contables de forma manual y en archivos de hojas de cálculo, por lo que se dificultaba realizar un

análisis en tiempo real. La información se exponía al error humano por existir procesos manuales como la constatación de existencias, el cruce de valores entre lo que el sistema contable reflejaba y los egresos que el encargado de la bodega de suministros contabilizaba mediante los comprobantes de egreso sin tener en cuenta la fecha ni precios de compra de los productos y, financieramente esto repercutía en los valores que dichos egresos generaban.

Mediante el sistema web con su módulo de gestión de bodega, se permite optimizar el manejo de información garantizando su disponibilidad en tiempo real de una forma segura.

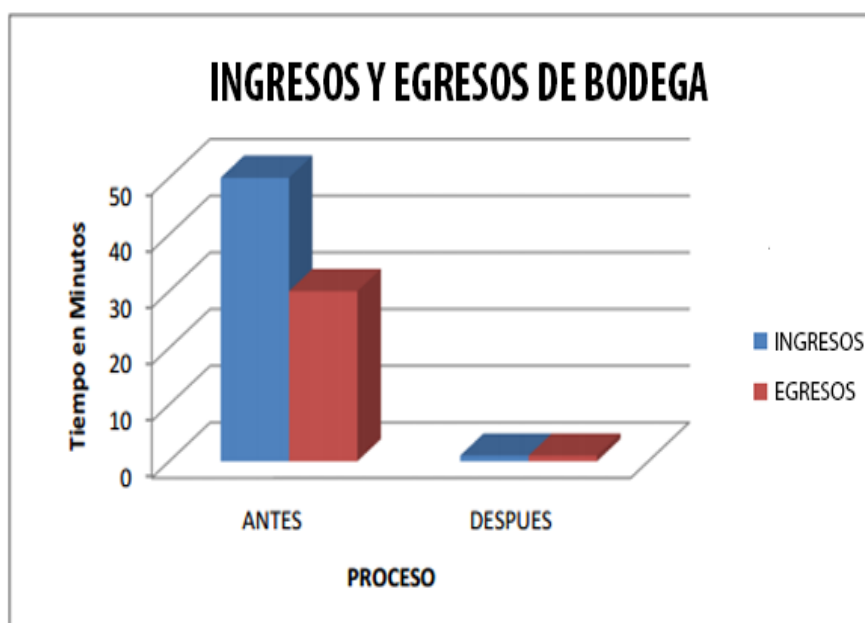


FIGURA 6.1 ANÁLISIS DE IMPACTO MÓDULO BODEGA

Como podemos apreciar, el gráfico anterior describe el tiempo que tomaba sacar un reporte contable de la mercadería existente en bodega y el tiempo que toma actualmente obtener un reporte financiero del estado contable de existencias en bodega. Antes de la implantación del sistema se demoraba aproximadamente 40 minutos en obtener un reporte financiero o balance de los movimientos de materiales e insumos en bodega, por lo que después de la implantación del sistema se obtiene mencionada información en 5 minutos aproximadamente.

6.1.2 Asignación de Licencias y Permisos.-

La planificación de Licencias y Permisos se demoraba aproximadamente 30 días, ya que de forma manual, el amanuense o secretario de la división de recursos humanos realizaba un cronograma anual de licencias para todos los empleados de la organización.

Conforme se iban incrementando los empleados enlistados se iban generando problemas en cuanto a las fechas para las licencias planificadas debido a la falta de un sistema que dosifique adecuadamente las licencias para todos los meses del año optimizando el recurso humano y garantizando la operatividad de la organización en cada una de sus dependencias y retenes.

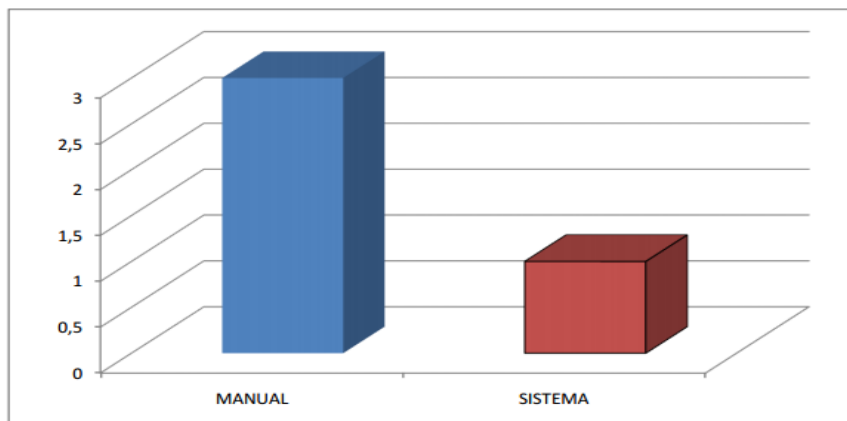


FIGURA 6.2 ANÁLISIS DE IMPACTO LICENCIAS Y PERMISOS

En la figura anterior, se refleja la comparativa del tiempo que toma una adecuada planificación de las licencias para el personal de la organización.

Con la implementación del sistema se puede obtener un plan anual de licencias en no menos de 2 días por cuanto cada empleado puede acceder en tiempo real y planifica su licencia, además que el usuario supervisor de recursos humanos tiene la herramienta de visualizar al personal disponible en cada subdivisión de la organización antes de aprobar una licencia planificada. Del mismo modo la aprobación de una licencia planificada puede tomar 10 minutos tomando en cuenta que los empleados se encuentran en lugares apartados de la DIRGES y que solo lo hacen vía web evitándoles tener que movilizarse hasta Esmeraldas para tal efecto.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

1. La implementación del sistema informático para la Gestión de Recursos humanos y para la Gestión de Insumos en los repartos navales pertenecientes a la DIRGES, permitirá optimizar recursos llevando de manera ordenada y sistematizada la administración del talento humano y la distribución de materiales e insumos en dichos retenes.
2. La plataforma utilizada es vía web, lo que permitirá el acceso a la misma desde cualquier punto remoto con acceso al internet descentralizando la administración de recursos.

3. La tecnología que se utilizó en el desarrollo de este proyecto facilita mucho a los programadores para llevar el código fuente de una manera ordenada, cómoda y segura.
4. Se comprobó que la utilización de las interfaces OpenSource manejan de manera óptima los recursos Ajax, lo que permite que la aplicación sea mucho más eficiente como también ayuda en el tiempo de desarrollo ya que existen los componentes predefinidos y personalizables acorde a las particularidades de la empresa o institución.
5. Se aprovechó la página inicial de la aplicación para publicar información referente a las actividades de la DIRGES desarrolla, lo que se constituyó en un valor agregado para la misma.
6. El diseño de la aplicación es de fácil y sencillo manejo, lo que disminuye el riesgo de errores humanos que puedan ocasionar pérdida de información.

RECOMENDACIONES

1. Para garantizar un óptimo funcionamiento de la aplicación DIRGES se requiere un paquete preparado wamp para Windows, o xampp para Linux y Mac, un servidor Apache 1.3 o superior o Microsoft Internet Information Services (IIS), PHP 5.2 o superior instalado y habilitado y un servidor de base de datos MySQL 5.0 o superior.

2. Se recomienda un navegador web Internet Explorer 10 o superior, Mozilla Firefox 36 o superior. Al ser una aplicación responsive se ajustará dicho contenido al dispositivo que se utilice para su visualización.
3. Se debe usar las herramientas de validación de código que tienen los entornos de desarrollo de software, estas herramientas ayudan a mejorar el estilo de programación.
4. Se recomienda para el desarrollo de futuros módulos usar arquitectura de n capas que permitan la optimización del tiempo de construcción y también faciliten el mantenimiento de las aplicaciones.
5. Se recomienda a las Autoridades de la Dirección Nacional de los Espacios Acuáticos, analizar la factibilidad de habilitar en cada retén naval perteneciente a las direcciones regionales, un punto de acceso a la intranet institucional a fin de incrementar la seguridad de las aplicaciones web, así como también disminuir el tiempo de respuesta en las transacciones en línea que se realicen.

BIBLIOGRAFÍA

- [1] D.V.S.Sosa, «http://www.tamps.cinvestav.mx/~vjsosa/clases/sd/Middleware_Recorrido.pdf,» 3 Marzo 2014. [En línea]. Available: <http://www.tamps.cinvestav.mx>. [Último acceso: 3 Agosto 2016].
- [2] M. J. M. Galindo y J. P. Blasquez, Escaneando la Informática, Barcelona: UOC, 2010.
- [3] J. M. Caballero, Implantación de Aplicaciones Informáticas de Gestión., Madrid: Vision NET, 2010.
- [4] J. J. R. Lara, Introducción a la Programación Teoría y Práctica, Alicante: Club Universitario, 2003.
- [5] M. Pérez, LENGUAJES de PROGRAMACION ORIENTADA a OBJETOS, Createspace Independent Pub, 2014.
- [6] I. Sommerville, Ingeniería de Software, Madrid: Pearson Educación, 2005.
- [7] R. A. Villamarín Yagual, «<http://repositorio.espe.edu.ec/handle/21000/7180>,» Julio 2013. [En línea]. Available: www.espe.edu.ec. [Último acceso: 12 Agosto 2016].
- [8] jordisan, «<http://jordisan.net/blog/2006/que-es-un-framework/>,» 29 Septiembre 2006. [En línea]. [Último acceso: 22 Agosto 2016].
- [9] Jean, «<http://www.catswhocode.com/blog/top-10-php-frameworks-for-2014>,» 9 junio 2014. [En línea]. [Último acceso: 30 Agosto 2016].
- [10] M. G. P. M. G. A. & M. L. Díaz-Antón, Propuesta de una metodología de desarrollo de software educativo bajo un enfoque de calidad sistémica, Caracas: Univ. Simón Bolívar, 2006.
- [11] jummpsblog, «<https://jummp.wordpress.com/about/>,» 6 Abril 2011. [En línea]. [Último acceso: 1 Septiembre 2016].
- [12] DefinicionABC, «<http://www.definicionabc.com/economia/inventario.php>,» 20 Enero 2016. [En línea]. [Último acceso: 11 Agosto 1026].
- [13] Blogger, «<http://yanetcedillor.blogspot.com/p/servidores-web.html>,» 22 Febrero 2015. [En línea]. [Último acceso: 12 Julio 2016].
- [14] H. Encinas, «<http://hugoencinas.weebly.com/blog/lenguajes-de-programacin>,» 1

Mayo 2014. [En línea]. [Último acceso: 22 Marzo 2016].

[15] J. E. R. Sánchez, «<http://enriqueperudata.blogspot.com/2013/08/mysql-server-parte-1.html>,» 5 Agosto 2013. [En línea]. [Último acceso: 22 Agosto 2016].

[16] Wikipedia, «https://es.wikipedia.org/wiki/Programaci%C3%B3n_por_capas,» 23 Febrero 2013. [En línea]. [Último acceso: 11 Agosto 2016].