



UNIVERSIDAD TÉCNICA DE BABAHOYO

FACULTAD DE ADMINISTRACION, FINANZAS E INFORMATICA

F.A.F.I

Proyecto de Tesis

Previo a la Obtención del Título de:

Ingeniero en Sistemas

Tema:

**DESARROLLO DE UN SISTEMA DE GESTIÓN DE ANÁLISIS CLÍNICO,
PARA MEJORAR EL CENTRO DE SALUD DE ECHEANDIA.**

Proponentes:

Edwin Juvenal Escobar Carvajal

Freddy Gabriel Muñoz Oviedo

DIRECTOR DE TESIS

Ing. María Isabel Gonzales Valero

LECTOR DE TESIS:

Ing. Ángel Rafael España León



UNIVERSIDAD TECNICA DE BABAHOYO

Facultad de Administración, Finanzas e Informática

F.A.F.I.

Escuela de Sistemas

DECLARACIÓN DE AUTORÍA.

Ante las autoridades de la Facultad de Administración Finanzas e Informática de la Universidad Técnica de Babahoyo declaro que el contenido del trabajo de grado cuyo título es “DESARROLLO DE UN SISTEMA DE GESTIÓN DE ANÁLISIS CLÍNICO, PARA MEJORAR EL CENTRO DE SALUD DE ECHEANDIA”, presentado como requisito para la obtención del título de Ingeniero en Sistemas es original, de mi autoría y total responsabilidad.

Atentamente,

Edwin Juvenal Escobar Carvajal

Freddy Gabriel Muñoz Oviedo



UNIVERSIDAD TECNICA DE BABAHOYO

Facultad de Administración, Finanzas e Informática

F.A.F.I.

Escuela de Sistemas

CERTIFICACIÓN.

En calidad de asesor del presente trabajo de investigación, certifico que el trabajo de grado cuyo título es “**DESARROLLO DE UN SISTEMA DE GESTIÓN DE ANÁLISIS CLÍNICO, PARA MEJORAR EL CENTRO DE SALUD DE ECHEANDIA**”, Fue elaborado por los egresados: Edwin Juvenal Escobar Carvajal y Freddy Gabriel Muñoz Oviedo, y cumplen con los requisitos metodológicos y científicos que la Universidad Técnica de Babahoyo exige, por lo tanto autorizo su presentación para los trámites pertinentes.

Atentamente:

Asesor.



UNIVERSIDAD TECNICA DE BABAHOYO

Facultad de Administración, Finanzas e Informática

F.A.F.I.

Escuela de Sistemas

AGRADECIMIENTO.

Ante todo agradezco a Dios por haberme dado la oportunidad de existir, y por darme sabiduría y el conocimiento necesario para poder llevar a cabo la realización de este proyecto de Tesis, a mis padres quienes me motivaron psicológicamente y económicamente, dándome el respectivo apoyo para superar cualquier inconveniente, problema, adversidad, que pudo a ver surgido en el transcurso que dura la carrera de Ingeniería en sistemas.

El deseo de superación ha hecho posible mantenerme firme y con ganas de aprender cada día cosas nuevas y con el objetivo de cumplir cada una de mis metas que me he propuesto.

Agradezco como directora de tesis a la Ing. María Isabel Gonzales Valero y como lector de tesis al Ing. Ángel España León que me ha influido el conocimiento necesario, por su bondad y experiencia, por su valioso tiempo y orientaciones en este proyecto de Tesis.

Freddy Gabriel Muñoz Oviedo



UNIVERSIDAD TECNICA DE BABAHOYO

Facultad de Administración, Finanzas e Informática

F.A.F.I.

Escuela de Sistemas

DEDICATORIA

Esta tesis se lo dedico en especial a Dios y a mis queridos padres quienes con mucho esfuerzo y sacrificio me apoyaron en mi preparación profesional, ya que siempre, estuvieron con migo guiándome y animándome para nunca rendirme, ni darme por vencido para alcanzar mis metas, que con tanto sacrificio, esfuerzo, y trabajo, he luchado día, día para alcanzar esta meta propuesta.

También agradeciendo a mis queridos profesores: **Ingeniera María Isabel Gonzales y al Ing. Ángel España León** por su dedicación y apoyo en el aprendizaje que siempre me transmitieron para de esta manera lograr desarrollar mis grandes intelectos físicos y mentales, para poder ser un profesional y ser útil para la sociedad, y de esta manera poder cumplir con uno de mis sueños más grandes que es tener mi título de Ingeniero y poder ingresar al mundo profesional de la tecnología avanzada como lo es la Informática.

Freddy Gabriel Muñoz Oviedo



UNIVERSIDAD TECNICA DE BABAHOYO

Facultad de Administración, Finanzas e Informática

F.A.F.I.

Escuela de Sistemas

AGRADECIMIENTO

Expreso mi más profundo agradecimiento primeramente a Dios, que me ha colmado de bendiciones, paciencia, gozo, salud, amor a mis alrededores o a mis seres queridos por el aprecio y gran sabiduría, permitiendo de esta manera culminar una vez más otra etapa de mi carrera profesional, en área la de ingeniería, el control de calidad, la atención al cliente , mejorando el servicio e innovando la aplicación en la humanización de las empresas, siendo un emprendedor mas, pero simplemente todo se logra a través de una dirección sabia que viene de Dios por ejemplo de guía para llegar a la meta deseada y excelencia al éxito en todas sus áreas.

Agradeciendo también a mis padres quienes siempre me apoyaron mis sueños y aspiraciones en la vida y por la influencia de valores morales.

Esperando la satisfacción de mis padres por haberme convertido en un verdadero profesional y continuar siendo el hombre que ellos esperaban de mí, enorgulleciéndolos con cada una de mis acciones.

Agradezco al Ingeniero Ángel España, y la Ingeniera. María Isabel Gonzales Valero por su valiosa y oportuna ayuda prestada en nuestro trabajo de tesis. Gracias.

Edwin Juvenal Escobar Carvajal



UNIVERSIDAD TECNICA DE BABAHOYO

Facultad de Administración, Finanzas e Informática

F.A.F.I.

Escuela de Sistemas

DEDICATORIA

Esta tesis se lo dedico en especial a Dios y a mis queridos padres quienes con mucho esfuerzo y sacrificio me apoyaron en mi preparación profesional, luchado día, día para alcanzar esta meta propuesta ya que hubieron momentos tan difíciles pero siempre con la cabeza en alto gracias a la ayuda de Dios he llegado a culminar otra etapa de la vida profesional .

En segundo lugar a mis padres quienes han compartido cada caída y levantada que yo he tenido en el transcurso de la carrera, se que siempre habrán lindos recuerdos por haber compartido los conocimientos con los compañeros y profesores, agradezco rotundamente a mis padres por la inversión de mi carrera.

Edwin Juvenal Escobar Carvajal



UNIVERSIDAD TECNICA DE BABAHOYO
Facultad de Administración, Finanzas e
Informática
F.A.F.I.
Escuela de Sistemas

CERTIFICACIÓN DE AUTORÍA

La responsabilidad por los hechos, ideas y doctrinas expuestos en esta tesis me corresponden exclusivamente, y el patrimonio intelectual de la misma a la Escuela de Sistemas e Informática de la Universidad Técnica de Babahoyo.

Edwin Juvenal Escobar Carvajal

Freddy Gabriel Muñoz Oviedo



UNIVERSIDAD TECNICA DE BABAHOYO

Facultad de Administración, Finanzas e Informática

F.A.F.I.

Escuela de Sistemas

TRIBUNAL DE APROBACIÓN DE LA TESIS

Lcdo. Teodoro Flores Carpio

DECANO DE LA FACULTAD

Lcdo. Washington Ávila Ortega

DIRECTOR DE ESCUELA

Ing. María Isabel Gózales Valero

DIRECTORA DE TESIS

Ing. Ángel España León

LECTOR DE TESIS

Sr. Freddy Vásquez Bermeo

SECRETARIO



UNIVERSIDAD TECNICA DE BABAHOYO
Facultad de Administración, Finanzas e
Informática
F.A.F.I.
Escuela de Sistemas

TRIBUNAL DE SUSTENTACION

El Tribunal de Sustentación otorga al presente trabajo de investigación la calificación de..... Equivalente
a.....

Lcdo. Teodoro Flores Carpio

DECANO DE LA FACULTAD

Lcdo. Washington Ávila Ortega

DIRECTOR DE ESCUELA

Ing. María Isabel Gózales Valero

DIRECTORA DE TESIS

Ing. Ángel España León

LECTOR DE TESIS

Sr. Freddy Vásquez Bermeo

SECRETARIO



UNIVERSIDAD TECNICA DE BABAHOYO

Facultad de Administración, Finanzas e Informática

F.A.F.I.

Escuela de Sistemas

CARTA DE ACEPTACION DEL TUTOR

Comité de Trabajos de Grado

Fecha: _____

Escuela de Comunicación Social

Presente.-

Estimados señores:

Quien suscribe, Ing. _____

Hace constar que conoce el proyecto correspondiente y ha aceptado formalmente la tutoría del Trabajo de Grado que tiene por título:

Con el cual el (los) estudiante(s):

	Nombre y apellido	Expediente	Mención
1	Edwin Juvenal Escobar Carvajal		
2	Freddy Gabriel Muñoz Oviedo		

de X semestre de Ingeniero en Sistemas, aspira(n) obtener el título de Ingeniero(s) en Sistemas. Declaro también conocer y aceptar las Disposiciones de la escuela de Sistemas sobre Trabajos de Grado, así como el calendario que registrará las actividades de seguimiento de tesis a lo largo del presente año académico.

Firma del Tutor



UNIVERSIDAD TECNICA DE BABAHOYO
Facultad de Administración, Finanzas e
Informática
F.A.F.I.

Escuela de Sistemas
CARTA DE ACEPTACION DEL TUTOR
FIRMAS DE PERSONALIDAD

El jurado calificador de la **ESCUELA DE SISTEMAS-FACULTAD DE ADMINISTRACIÓN FINANZAS E INFORMÁTICA** de la **UNIVERSIDAD TÉCNICA DE BABAHOYO** le da al siguiente proyecto de tesis.

La calificación de _____

Equivalente a _____

Fecha: _____

Firma para corroborar su veracidad:

Presidente Tribunal de Defensa _____

Director (a) de la Tesis _____

Lector de la Tesis _____

Secretario _____

INDICE GENERAL

Índice General.....	13
Índice De Cuadros.....	18
Índice De Gráficos.....	19
Resumen Ejecutivo-Summary.....	20
Introducción.....	22

CAPITULO I

El Problema.....	24
1.1 Problematización.....	24
1.1.1 Planteamiento del problema.....	24
1.1.2 Formulación del problema.....	24
1.1.3 Preguntas de investigación.....	24
1.1.4 Delimitación del problema.....	25
1.1.5 Objetivos de la investigación.....	26
1.1.5.1 Objetivo General.....	26
1.1.5.2Objetivos Específicos.....	26
1.1.6 Justificación de la Investigación.....	26
1.1.7 Viabilidad Operativa.....	27
1.1.7.1Sistemas Informáticos.....	27
1.1.7.2 Software Libre.....	28
1.1.7.2.1 ¿Qué es software libre?.....	28
1.1.7.3 Clasificación del Lenguaje de Programación.....	30
1.1.7.3.1 HTML.....	35
1.1.7.3.2 Historia de HTML.....	36
1.1.7.3.3 Marcado HTML.....	36

1.1.7.3.3.1 Elementos.....	36
1.1.7.3.3.2 Atributos.....	38
1.1.7.3.3.3 Códigos HTML (Básicos).....	38
1.1.7.3.3.3 Clases de Paginas WWW en lenguaje HTML.....	42
1.1.7.3.3.4 Tabulaciones.....	46
1.1.7.3.3.5 El HTML (HIPERTEXT MAKE UP LANGUAGE).....	48
1.1.7.3.3.6 Estructura de una Página.....	52
1.1.7.3.4 PHP.....	61
1.1.7.3.4.1 Introducción.....	61
1.1.7.3.4.1.1 ¿Qué es PHP?.....	61
1.1.7.3.4.2 Historia.....	62
1.1.7.3.4.3 ¿Qué se puede hacer con PHP?.....	64
1.1.7.3.5 Adobe Dreamweaver CS3.....	68
1.1.7.3.5.1 Introducción.....	68
1.1.7.3.5.2 Novedades de Dreamweaver.....	68
1.1.7.3.5.3 Entorno de trabajo.....	69
1.1.7.3.6 Base de Datos.....	73
1.1.7.3.6.1 Tipos de Base de Datos.....	74
1.1.7.3.6.2 Según el contenido.....	74
1.1.7.3.6.3 Modelo de Base de Datos.....	75
1.1.7.3.6.4 Lenguaje.....	80
1.1.7.3.6.5 Ventajas.....	80
1.1.7.3.6.6 Desventajas.....	81
1.1.7.3.6.7 Fases.....	81
1.1.7.3.6.8 Interpretación.....	81
1.1.7.3.6.9 Mecanismos.....	82
1.1.7.3.6.10 Conexiones de Base de Datos.....	82
1.1.7.3.7 ¿Qué es MySQL?.....	83
1.1.7.3.7.1 Características Principales de MySQL.....	84
1.1.7.3.7.2 Conectándose y Desconectándose al Servidor MySQL.....	85

1.1.7.3.8 Servidor Web Apache.....	86
1.1.7.3.9 El Protocolo HTTP.....	86
1.1.7.3.10 Arquitectura del Servidor Apache.....	87
1.1.7.3. 11 Seguridad de la Web.....	87
1.1.7.3. 12 Seguridad en la Transmisión.....	88
1.1.7.3.13 Instalación de SQL Yog 8.12.....	92

CAPITULO II

Marco Teórico.....	96
2.1 Antecedentes Y Estudio Previos.....	96
2.2 Fundamentación Teórica.....	96
2.2.1 Introducción.....	96
2.3 Definición De Términos.....	96
2.3.1 Estructuración De Centro De Salud.....	98
2.3.1.1 Ubicación Geográfica.....	98
2.3.1.2 Relación De Los Antecedentes Y Transcendencias de la Entidad.....	99
2.3.2 Estructura Organización Por Proceso.....	100
2.3.3 Obtener Manuales de Procedimientos Respecto a las Actividades.....	100
2.3.4 Funcionamiento del Laboratorio.....	101
2.3.5 Estudio De Misión y Visión del Laboratorio.....	102
2.3.5.1 Objetivos Estratégicos.....	103
2.3.6 Estructura Orgánica.....	104
2.3.7 Entrada del centro de salud.....	105

CAPITULO III

Marco Hipótesis.....	109
3.1 Planteamientos de Hipótesis y Variables.....	109
3.1.1 Planteamiento de Hipótesis.....	109
3.1.2 Planteamiento de Variables.....	109
3.2 Operacionalización de Variables e Indicadores.....	109
3.2.1 Hipótesis.....	109
3.2.2 Variables.....	109

CAPITULO IV

Marco Metodológico.....	111
4.1 Diseño De La Investigación.....	111
4.1.1 Modalidad De La Investigación.....	111
4.1.2 Tipo De La Investigación.....	112
4.1.3 Población Y Muestra.....	113
4.1.4 Métodos, Técnicas e Instrumentos.....	113
4.2 Procesamiento y Análisis.....	125
4.2.1 Diagrama de Caso de Uso.....	125
4.2.2 Diagrama de Secuencia.....	126
4.2.3 Diagrama de Actividad	127
4.2.4 Diagrama de Despliegue.....	128
4.2.5 Diagrama de Entidad-Relación.....	129
4.2.6 Diccionario de Datos.....	130
4.3 Procesamiento de la Información.....	138
4.4 Análisis e Interpretación de Resultados.....	138
4.5 Conclusiones y Recomendaciones.....	140
4.5.1 Conclusiones.....	140
4.5.2 Recomendaciones.....	141

CAPITULO V

Propuesta.....	143
5.1 Titulo De La Propuesta.....	143
5.2 Objetivos.....	143
5.2.1 Objetivo General.....	143
5.2.2 Objetivo Especifico.....	143
5.3 Justificación.....	144
5.4 Importancia.....	145
5.5 Viabilidad.....	145
5.5.1 Viabilidad Técnica.....	145
5.5.2 Viabilidad Económica.....	145
5.5.3 Viabilidad Operativa.....	146
5.6 Fundamentación Teórica.....	146
5.7 Descripción de la Propuesta.....	147
5.8 Ejecución de la Propuesta.....	148
5.9 Presupuesto.....	149
5.10 Cronograma Tentativo.....	150
5.11 Evaluación.....	151
5.12 Impacto.....	151
5.12.1 Beneficios del Sistema.....	151
5.12.2 Económico.....	151
5.12.3 Seguro.....	152
Conclusiones.....	153
Recomendaciones.....	154
Glosario de Términos y Abreviaturas.....	155
Referencias Bibliográficas.....	158
Anexos.....	159

INDICE DE CUADROS

Tabla Nº 1: Comandos usados en HTML.....	52
Tabla Nº 2: Etiqueta en HTML.....	56
Tabla Nº 3: Estructura Organizacional Por Proceso.....	100
Tabla Nº 4: Estructura Orgánica.....	104
Tabla Nº 5: Pantalla del Diseño de la Investigación.....	111
Tabla Nº 6: Encuesta Primera Pregunta.....	115
Tabla Nº 7: Encuesta Segunda Pregunta.....	116
Tabla Nº 8: Encuesta Tercera Pregunta	117
Tabla Nº 9: Encuesta Cuarta Pregunta	118
Tabla Nº 10: Encuesta Quinta Pregunta	119
Tabla Nº11: Encuesta Sexta Pregunta	120
Tabla Nº 12: Encuesta Séptima Pregunta	121
Tabla Nº 13: Encuesta Octava Pregunta.....	122
Tabla Nº 14: Encuesta Novena Pregunta	123
Tabla Nº 15: Encuesta Decima Pregunta.....	124
Tabla Nº 16: Entidad- Relación del Sistema del Laboratorio Clínico....	129
Tabla Nº 17: Proceso de Administrador.....	130
Tabla Nº 18: Proceso del Doctor.....	131
Tabla Nº 19: Proceso de la Enfermera.....	131
Tabla Nº 20: Presupuesto.....	149
Tabla Nº 21: Cronograma Tentativo.....	150

INDICE DE GRAFICOS

Grafico Nº1: Pantalla Adobe Dreamweaver	70
Grafico Nº 2: Dream administrar sitios	71
Grafico Nº 3: Ubicación de la carpeta.....	72
Grafico Nº 4: Instalación SQLYog.....	92
Grafico Nº 5: Instalación SQLYog.....	93
Grafico Nº 6: Instalación SQLYog.....	93
Grafico Nº 7: Instalación SQLYog.....	94
Grafico Nº 8: Instalación SQLYog.....	94
Foto Nº1 del Centro de Salud de Echeandía.....	105
Foto Nº2 Llegad Principal de Centro de Salud de Echeandía.....	105
Foto Nº3 Pasillo al Laboratorio.....	106
Foto Nº4 Sala del Laboratorio	106
Foto Nº5 Oficina del Laboratorio.....	107
Foto Nº6 Oficina del Laboratorio.....	107
Foto Nº7 Oficina del Laboratorio.....	107
Grafico Nº 9 Encuesta; primera pregunta	115
Grafico Nº 10 Encuesta; segunda pregunta.....	116
Grafico Nº 11 Encuesta; tercera pregunta.....	117
Grafico Nº 12 Encuesta; cuarta pregunta.....	118
Grafico Nº 13 Encuesta; quinta pregunta.....	119
Grafico Nº 14 Encuesta; sexta pregunta.....	120
Grafico Nº 15 Encuesta; séptima pregunta.....	121
Grafico Nº 16 Encuesta; octava pregunta.....	122
Grafico Nº 17 Encuesta; novena pregunta.....	123
Grafico Nº 18 Encuesta; decima pregunta.....	124
Gráfico Nº 19 Diagramas de Caso de Uso.....	125
Grafico Nº 20 Diagrama de Secuencia.....	126
Grafico Nº 21 Diagramas De Actividad.....	127
Grafico Nº 22 Diagramas De Despliegue.....	128

Resumen Ejecutivo- Summary

Introducción

Sistema de información es un conjunto de elementos orientados al tratamiento y administración de datos e información, para cubrir una necesidad u objetivo.

Los sistemas de información actualmente están siendo aceptados y adoptados por muchas empresas debido a las ventajas competitivas y a la optimización de los procesos en los cuales intervienen.

El modelo Cliente/Servidor reúne las características necesarias para proveer esta infraestructura, independiente del tamaño y complejidad de las operaciones de las organizaciones públicas o privadas.

Planteamiento del Problema

Al visitar el Centro Salud de la ciudad de Echeandía, se dio un profundo análisis de la problemática del laboratorio de examen, durante el análisis nos hemos dado cuenta de que lleva la información a través de registros, la cual no contiene un software en que me permita ayudar a archivar la información de manera rápida y eficaz y dar un resultado de la misma de acuerdo a la petición del paciente.

En vista de ciertas pérdidas y anomalías que han ocurrido y la incomodidad que causa al paciente el trámite del papeleo; y el tiempo que tarda.

Durante el análisis y diagnóstico mencionado anteriormente la población percibe una visión negativa; ya que en el transcurso del tiempo, viendo la necesidad por ello establecer alternativas que conlleve a mejorar en Centro Salud de la ciudad Echeandía.

Objetivo General

Desarrollar un sistema de gestión de análisis clínico, para mejorar el Centro de Salud de la Ciudad de Echeandía.

Justificación

Los sistemas de información del laboratorio clínico (SIL o LIS, como se ha denominado en capítulos previos) constituyen hoy en día una herramienta indispensable y crítica para la actividad de los laboratorios clínicos.

La evolución de los SIL en los últimos 25 años, unida al desarrollo de los grandes autos analizadores y la robótica, han posibilitado un aumento extraordinario de la capacidad y la eficiencia de los laboratorios con elevados niveles de calidad.

Desde los primitivos sistemas que simplemente almacenaban los datos del laboratorio para su impresión, los SIL han evolucionado hasta potentes sistemas que abarcan todas las facetas del laboratorio. En el apartado asistencial, los SIL ayudan en los aspectos preanalíticos (solicitud, cita, obtención de muestras, preparación, transporte, fraccionamiento y distribución), en los aspectos analíticos (procesamiento, gestión de equipos y rutas, control de calidad, validación técnica) y en los aspectos postanalíticos (validación clínica, edición de informes, distribución, archivo de muestras). Además, el SIL debe contemplar aspectos de los laboratorios con funcionalidades y requerimientos específicos como la microbiología, las urgencias, la genética, los análisis junto al paciente.

Junto con el aspecto asistencial, el SIL debe incorporar herramientas de gestión y comunicación (aprovisionamiento y gestión de almacenamiento de la información, tanto a nivel de gestión como a nivel científico y epidemiológico).

El SIL, con todas sus peculiaridades, debe integrarse en el conjunto de los sistemas de información asistencial potenciando aún más su utilidad.

Introducción

Sistema de información es un conjunto de elementos orientados al tratamiento y administración de datos e información, organizados y listos para su posterior uso, generados para cubrir una necesidad u objetivo. Dichos elementos formarán parte de alguna de estas categorías: personas, datos y actividades o técnicas de trabajo.

Recursos materiales en general típicamente recursos informáticos y de comunicación, aunque no tienen por qué ser de este tipo obligatoriamente.

Todos estos elementos interactúan entre sí para procesar los datos incluyendo procesos manuales y automáticos dando lugar a información más elaborada y distribuyéndola de la manera más adecuada posible en una determinada organización en función de sus objetivos.

Los sistemas de información actualmente están siendo aceptados y adoptados por muchas empresas debido a las ventajas competitivas que proporcionan y a la optimización de los procesos en los cuales intervienen.

La información es guardada según una estructura bien diseñada con el fin de tener acceso fácil y rápido a ella en cualquier momento. Los procesos alrededor del sistema de información se tornarán más claros pues este será un gestor que sigue unas reglas claras y ágiles.

El modelo Cliente/Servidor reúne las características necesarias para proveer esta infraestructura, independiente del tamaño y complejidad de las operaciones de las organizaciones públicas o privadas y, consecuentemente desarrolla un pale importante el proceso de la evaluación.

CAPITULO I

El Problema

1.1 Problematicación

1.1.1 Planteamiento del problema

Al visitar el Centro Salud de la ciudad de Echeandía, se dio un profundo diagnóstico y análisis de la problemática del laboratorio de examen, durante el análisis nos hemos dado cuenta de que lleva la información a través de registros, en el cual la información que se registra es un problemática o un desorden, la cual no contiene un software en que me permita ayudar a archivar la información de manera rápida y eficaz y dar un resultado de la misma de acuerdo a la petición del paciente.

Motivos por la cual se ve la necesidad de realizar una investigación profunda y minuciosa para ver en qué áreas se las puede mejorar en este Centro Salud.

En vista de ciertas perdidas y anomalías que han ocurrido a través del ingreso manual de cada registro de los pacientes y la incomodidad que causa al paciente el trámite del papeleo ; y el tiempo que tarda, en ingresar los datos de manera manualmente por cada paciente.

Durante el análisis y diagnóstico mencionado anteriormente la población percibe una visión negativa; ya que en el transcurso del tiempo, viendo la necesidad por ello establecer alternativas que conlleve a mejorar en Centro Salud de la ciudad Echeandía.

1.1.2 Formulación del problema

¿De qué manera se puede gestionar el análisis cíclico a través de un sistema informático?

1.1.3 Preguntas de investigación

1. ¿Sería necesario la capacitación informática en caso de implementar un sistemas informático?

- 2.- ¿Considera usted que hay bastante afluencia en el laboratorio clínico?
- 3.- ¿Cree usted que hay un desorden al momento de ingresar los registros de los pacientes?
- 4.- ¿Cree usted que hay una demora al momento de retirar los exámenes del laboratorio?
- 5.- ¿Considera usted que debido al manejo manual de la documentación las actividades del laboratorio se torna muy lentas?
- 6.- ¿Considera usted que se maneja gran cantidad de registro del laboratorio?
- 7.- ¿Cree usted que con un sistema adecuado se ordenaría de una mejor forma los registros del laboratorio?
- 8.- ¿Cree usted que un sistema de informático ayudaría a mejorar la atención al paciente?
- 9.- ¿Cree usted que mediante un sistema web mejoraría el tiempo de respuesta?
- 10.- ¿Está usted de acuerdo que se registre los análisis en un sistema informático para mejorar la agilidad del laboratorista?

1.1.4 Delimitación del problema

Objeto de estudio: Ingeniería en Sistemas.

Campo de acción: Sistema de Información.

Esta investigación se realizará en el Centro de Salud de la ciudad de Echeandía prov. Bolívar, ubicada en la avenida 5 DE OCTUBRE durante el año 2012.

1.1.5 Objetivos de la investigación

1.1.5.1 OBJETIVO GENERAL

Desarrollar un sistema de gestión de análisis clínico, para mejorar el Centro de Salud de la Ciudad de Echeandía.

1.1.5.2 OBJETIVO ESPECIFICO

- ✓ Fundamentar de bases teóricas y científicas que permitan el desarrollo de esta investigación.
- ✓ Analizar y preparar información para conocer las mejores soluciones.
- ✓ Validar información y resultados con la ayuda de un experto.
- ✓ Implementar una solución informática usando tecnología PHP.

1.1.6 Justificación de la Investigación

En la presente investigación y análisis está encaminada para mejorar el Centro de Salud ante la comunidad, proporcionando las posibles soluciones a la problemática que enfrenta dicha Institución pública de salud.

El sistema permite optimizar el registro de manera rápida y eficaz, todos los registros de los pacientes, en donde que contenga una información de un resultado final. Por medio de este sistema mejora el tiempo de respuesta de acuerdo a la necesidad del laboratorista como: guardar registro, modificar registro, elimina registro, consultas general, consultas específicas, reportes, ayuda de una manera estructurada del sistema de manera rápida, y eficaz. Este software, permite automatizar todos los registros de cada paciente de acuerdo a los análisis del examen, por medio de este sistema ayudaría a la comunidad para lograr un buen desarrollo y adquirir una buena imagen a la ciudadanía, para mejorar y brindar la atención al paciente.

1.1.7 Viabilidad Operativa

1.1.7 .1 Sistemas Informáticos

Informática: El término 'informática' proviene de la fusión de los términos "información" y "automática", y se define como la ciencia que estudia el tratamiento automático y racional de la información, como soporte de los conocimientos y comunicaciones humanas, llevado a cabo mediante elementos automáticos, así como el conjunto de técnicas, métodos y máquinas aplicadas a dicho tratamiento. La Real Academia Española de la Lengua da la siguiente definición: "Conjunto de conocimientos científicos y técnicas que hacen posible el tratamiento automático de la información por medio de computadoras electrónicas". De esta última definición podemos deducir que hay tanto una ciencia informática como unas técnicas informáticas.

Cuando se ocupa de la información como materia esencial de estudio, con esta información es preciso: representarla en forma eficiente y automatizable, transmitirla sin errores ni pérdidas, y almacenarla para poder acceder a ella y recuperarla tantas veces como sea preciso procesarla para obtener nuevas informaciones más elaboradas y más útiles a nuestros propósitos

Sistemas informáticos: Conjunto de elementos interconectados o relacionados para el tratamiento de información. El más básico es un ordenador típico. Los más complejos son las redes, sistemas de procesamiento en paralelo. En este término suelen incluirse los elementos físicos, el software, y otros relacionados. Así el contenido en "hardware" de una habitación en la que se encuentra instalado un ordenador también se puede denominar "el sistema", es decir, amplios elementos fabricados por distintas empresas y con diferentes cometidos. Suele contener este término el software básico, tal como el sistema operativo, y los compiladores, depuradores, enlazadores. También un conjunto de programas dedicados a una aplicación específica, lo que incluye este

software, el básico, los soportes, la documentación, manuales de procedimiento.

Información: Comunicación o adquisición de conocimientos que permiten ampliar o precisar los que se poseen sobre una materia determinada. Podría entenderse que si no se consigue alguna de las dos finalidades señaladas, no habría información, pero es prácticamente imposible que no concurra alguna de ellas cuando un ser humano se encuentra ante una exposición de conocimientos. Para que la información sea la adecuada se tendrán que cumplir unos cuantos requisitos: precisión, significatividad.

Un lenguaje de programación es un lenguaje inventado para controlar una máquina, (normalmente, un ordenador). Hay muchísimos, de toda clase de tipos y características, inventados para facilitar el abordaje de distintos problemas, el mantenimiento del software, su reutilización, mejorar la productividad.

1.1.7 .2 Software Libre

1.1.7.2.1 ¿Qué es el Software Libre?

Software Libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. De modo más preciso, se refiere a cuatro libertades de los usuarios del software:

- La libertad de usar el programa, con cualquier propósito.
- La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades.
- El acceso al código fuente es una condición previa para esto.
- La libertad de distribuir copias.
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie.
- El acceso al código fuente es un requisito previo para esto.

Un programa es software libre si los usuarios tienen todas estas libertades. Así pueden, tener la libertad de distribuir copias, sea con o sin modificaciones, sea gratis o cobrando una cantidad por la distribución, a cualquiera y a cualquier lugar. El ser libre de hacer esto significa (entre otras cosas) que no tienes que pedir o pagar permisos.

También pueden tener la libertad de hacer modificaciones y utilizarlas de manera privada en tu trabajo u ocio, sin ni siquiera tener que anunciar que dichas modificaciones existen. Si publican tus cambios, no tienes por qué avisar a nadie en particular, ni de ninguna manera en particular.

La libertad para usar un programa significa la libertad para cualquier persona u organización de usarlo en cualquier tipo de sistema informático, para cualquier clase de trabajo, y sin tener obligación de comunicárselo al desarrollador o a alguna otra entidad específica.

‘Software libre’ no significa *‘no comercial’*. Un programa libre debe estar disponible para uso comercial, desarrollo comercial y distribución comercial. El desarrollo comercial del software libre ha dejado de ser inusual; el software comercial libre es muy importante.

1.1.7 .3 Clasificación de Lenguajes de Programación

Los lenguajes de programación se pueden clasificar según varios criterios. He encontrado doce en total: Nivel de abstracción, propósito, evolución histórica, manera de ejecutarse, manera de abordar la tarea a realizar, paradigma de programación, lugar de ejecución, concurrencia, interactividad, realización visual, determinismo y productividad.

Hay que tener en cuenta también, que en la práctica, la mayoría de lenguajes no pueden ser puramente clasificados en una categoría, pues surgen

incorporando ideas de otros lenguajes y de otras filosofías de programación, pero no importa al establecer las clasificaciones, pues el auténtico objetivo de las mismas es mostrar los rangos, las posibilidades y tipos de lenguajes que hay.

1. Nivel de abstracción

Según el nivel de abstracción, o sea, según el grado de cercanía a la máquina:

- Lenguajes de **bajo nivel**: La programación se realiza teniendo muy en cuenta las características del procesador. Ejemplo: Lenguajes ensamblador.
- Lenguajes de **nivel medio**: Permiten un mayor grado de abstracción pero al mismo tiempo mantienen algunas cualidades de los lenguajes de bajo nivel. Ejemplo: C
 - Puede realizar operaciones lógicas y de desplazamiento con bits, tratar todos los tipos de datos como lo que son en realidad a bajo nivel (números), etc.
- Lenguajes de **alto nivel**: Más parecidos al lenguaje humano. Manejan conceptos, tipos de datos, etc., de una manera cercana al pensamiento humano ignorando (abstrayéndose) del funcionamiento de la máquina. Ejemplos: Java, Ruby.

Hay quien sólo considera lenguajes de bajo nivel y de alto nivel, (en ese caso, C es considerado de alto nivel).

2. Propósito

Según el propósito, es decir, el tipo de problemas a tratar con ellos:

- Lenguajes de propósito **general**: Aptos para todo tipo de tareas: Ejemplo: C.

- Lenguajes de propósito **específico**: Hechos para un objetivo muy concreto. Ejemplo: Csound (para crear ficheros de audio).
- Lenguajes de **programación de sistemas**: Diseñados para realizar sistemas operativos o drivers. Ejemplo: C.
- Lenguajes de **script**: Para realizar tareas varias de control y auxiliares. Antiguamente eran los llamados lenguajes de procesamiento por lotes (batch) o JCL (“Job Control Languages”). Se subdividen en varias clases (de shell, de GUI, de programación web, etc.). Ejemplos: bash (shell), mIRC script, JavaScript (programación web).

3. Evolución histórica

Con el paso del tiempo, se va incrementando el **nivel de abstracción**, pero en la práctica, los de una generación no terminan de sustituir a los de la anterior:

- Lenguajes de **primera generación (1GL)**: Código máquina.
- Lenguajes de **segunda generación (2GL)**: Lenguajes ensamblador.
- Lenguajes de **tercera generación (3GL)**: La mayoría de los lenguajes modernos, diseñados para facilitar la programación a los humanos. Ejemplos: C, Java.
- Lenguajes de **cuarta generación (4GL)**: Diseñados con un propósito concreto, o sea, para abordar un tipo concreto de problemas. Ejemplos: NATURAL, Mathematica.
- Lenguajes de **quinta generación (5GL)**: La intención es que el programador establezca el qué problema ha de ser resuelto y las condiciones a reunir, y la máquina lo resuelve. Se usan en inteligencia artificial. Ejemplo: Prolog.

4. Manera de ejecutarse

Según la manera de ejecutarse:

- Lenguajes **compilados**: Un programa traductor traduce el código del programa (código fuente) en código máquina (código objeto). Otro programa, el enlazador, unirá los ficheros de código objeto del programa principal con los de las librerías para producir el programa ejecutable. Ejemplo: C.
- Lenguajes **interpretados**: Un programa (intérprete), ejecuta las instrucciones del programa de manera directa. Ejemplo: Lisp.

También los hay mixtos, como Java, que primero pasan por una fase de compilación en la que el código fuente se transforma en “bytecode”, y este “bytecode” puede ser ejecutado luego (interpretado) en ordenadores con distintas arquitecturas (procesadores) que tengan todos instalados la misma “máquina”.

5. Manera de abordar la tarea a realizar.

Según la manera de abordar la tarea a realizar, pueden ser:

- Lenguajes **imperativos**: Indican cómo hay que hacer la tarea, es decir, expresan los pasos a realizar. Ejemplo: C.
- Lenguajes **declarativos**: Indican qué hay que hacer. Ejemplos: Lisp, Prolog. Otros ejemplos de lenguajes declarativos, pero que no son lenguajes de programación, son HTML (para describir páginas web) oSQL (para consultar bases de datos).

6. Paradigma de programación

El **paradigma de programación** es el estilo de programación empleado. Algunos lenguajes soportan varios paradigmas, y otros sólo uno. Se puede

decir que históricamente han ido apareciendo para facilitar la tarea de programar según el tipo de problema a abordar, o para facilitar el mantenimiento del software, o por otra cuestión similar, por lo que todos corresponden a lenguajes de alto nivel (o nivel medio), estando los lenguajes ensambladores “atados” a la arquitectura de su procesador correspondiente. Los principales son:

- Lenguajes de **programación procedural**: Divide el problema en partes más pequeñas, que serán realizadas por subprogramas (subrutinas, funciones, procedimientos), que se llaman unas a otras para ser ejecutadas. Ejemplos: C, Pascal.
- Lenguajes de **programación orientada a objetos**: Crean un sistema de clases y objetos siguiendo el ejemplo del mundo real, en el que unos objetos realizan acciones y se comunican con otros objetos. Ejemplos: C++, Java.
- Lenguajes de **programación funcional**: La tarea se realiza evaluando funciones, (como en Matemáticas), de manera recursiva. Ejemplo: Lisp.
- Lenguajes de **programación lógica**: La tarea a realizar se expresa empleando lógica formal matemática. Expresa qué computar. Ejemplo: Prolog.

Hay muchos paradigmas de programación: Programación genérica, programación reflexiva, programación orientada a procesos, etc.

7. Lugar de ejecución.

En **sistemas distribuidos**, según dónde se ejecute:

- Lenguajes de **servidor**: Se ejecutan en el servidor. Ejemplo: PHP es el más utilizado en servidores web.

- Lenguajes de **cliente**: Se ejecutan en el cliente. Ejemplo: JavaScript en navegadores web.

8. Concurrencia.

Según admitan o no concurrencia de procesos, esto es, la ejecución simultánea de varios procesos lanzados por el programa:

- Lenguajes **concurrentes**. Ejemplo: Ada.
- Lenguajes **no concurrentes**. Ejemplo: C.

9. Interactividad.

Según la interactividad del programa con el usuario u otros programas:

- Lenguajes **orientados a sucesos**: El flujo del programa es controlado por la interacción con el usuario o por mensajes de otros programas/sistema operativo, como editores de texto, interfaces gráficos de usuario (GUI) o kernels. Ejemplo: Visual Basic, lenguajes de programación declarativos.
- Lenguajes **no orientados a sucesos**: El flujo del programa no depende de sucesos exteriores, sino que se conoce de antemano, siendo los procesos batch el ejemplo más claro (actualizaciones de bases de datos, colas de impresión de documentos, etc.). Ejemplos: Lenguajes de programación imperativos.

10. Realización visual.

Según la realización visual o no del programa:

- Lenguajes de **programación visual**: El programa se realiza moviendo bloques de construcción de programas (objetos visuales) en un interfaz adecuado para ello. No confundir con entornos de programación visual, como Microsoft Visual Studio y sus lenguajes de programación textuales (como Visual C#). Ejemplo: Mindscript.
- Lenguajes de **programación textual**: El código del programa se realiza escribiéndolo. Ejemplos: C, Java, Lisp.

11. Determinismo.

Según se pueda predecir o no el siguiente estado del programa a partir del estado actual:

- Lenguajes **deterministas**. Ejemplos: Todos los anteriores.
- Lenguajes **probabilísticos** o **no deterministas**: Sirven para explorar grandes espacios de búsqueda, (como gramáticas), y en la investigación teórica de hipercomputación. Ejemplo: mutt (generador de texto aleatorio).

1.1.7 .3.1 Html

HTML, siglas de HyperText Markup Language («lenguaje de marcado de hipertexto»), hace referencia al lenguaje de marcado predominante para la elaboración de páginas web que se utiliza para describir y traducir la estructura y la información en forma de texto, así como para complementar el texto con objetos tales como imágenes. El HTML se escribe en forma de «etiquetas», rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo JavaScript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.1

HTML también sirve para referirse al contenido del tipo de MIME text/html o todavía más ampliamente como un término genérico para el HTML, ya sea en forma descendida del XML (como XHTML 1.0 y posteriores) o en forma descendida directamente de SGML (como HTML 4.01 y anteriores).

1.1.7.3.2 Historia de HTML

La primera descripción de HTML disponible públicamente fue un documento llamado HTML Tags (Etiquetas HTML), publicado por primera vez en Internet por Tim Berners-Lee en 1991. Describe 22 elementos que incluyen el diseño inicial y relativamente simple de HTML. Trece de estos elementos todavía existen en HTML 4.

Berners-Lee consideraba a HTML una ampliación de SGML, pero no fue formalmente reconocida como tal hasta la publicación de mediados de 1993, por la IETF, de una primera proposición para una especificación de HTML: el boceto Hypertext Markup Language de Berners-Lee y Dan Connolly, el cual incluía una Definición de Tipo de Documento SGML para definir la gramática. El boceto expiró luego de seis meses, pero fue notable por su reconocimiento de la etiqueta propia del navegador Mosaic usada para insertar imágenes sin cambio de línea, que reflejaba la filosofía del IETF de basar estándares en prototipos con éxito. De la misma manera, el boceto competidor de Dave Raggett HTML+ (Hypertext Markup Format) (Formato de Marcaje de Hipertexto), de finales de 1993, sugería estandarizar características ya implementadas, como las tablas.

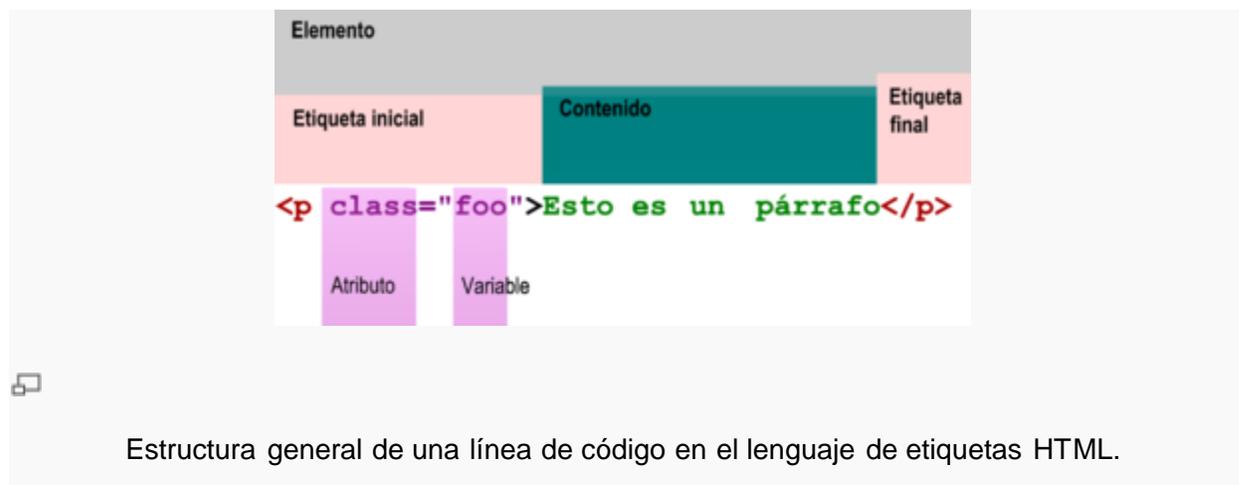
1.1.7.3.3 Marcado HTML

HTML consta de varios componentes vitales, entre ellos los elementos y sus atributos, tipos de data y la declaración de tipo de documento.

1.1.7.3.3.1 Elementos

Los elementos son la estructura básica de HTML. Los elementos tienen dos propiedades básicas: atributos y contenido. Cada atributo y contenido tiene ciertas restricciones para que se considere válido al documento HTML. Un elemento

generalmente tiene una etiqueta de inicio (por ejemplo, <nombre-de-elemento>) y una etiqueta de cierre (por ejemplo, </nombre-de-elemento>). Los atributos del elemento están contenidos en la etiqueta de inicio y el contenido está ubicado entre las dos etiquetas (por ejemplo, <nombre-de-elemento atributo="valor">Contenido</nombre-de-elemento>). Algunos elementos, tales como
, no tienen contenido ni llevan una etiqueta de cierre. Debajo se listan varios tipos de elementos de marcado usados en HTML.



El marcado *estructural* describe el propósito del texto. Por ejemplo, <h2>Golf</h2> establece «Golf» como un encabezamiento de segundo nivel, el cual se mostraría en un navegador de una manera similar al título «Marcado HTML» al principio de esta sección. El marcado estructural no define cómo se verá el elemento, pero la mayoría de los navegadores web han estandarizado el formato de los elementos. Puede aplicarse un formato específico al texto por medio de hojas de estilo en cascada.

El marcado *presentacional* describe la apariencia del texto, sin importar su función. Por ejemplo, negrita indica que los navegadores web visuales deben mostrar el texto en negrita, pero no indica qué deben hacer los navegadores web que muestran el contenido de otra manera (por ejemplo, los que leen el texto en voz alta). En el caso de negrita e <i>itálica</i>, existen elementos que se ven de la misma manera pero tienen una naturaleza más semántica: énfasis fuerte y énfasis. Es fácil ver cómo un lector de pantalla debería

interpretar estos dos elementos. Sin embargo, son equivalentes a sus correspondientes elementos presentacionales: un lector de pantalla no debería decir más fuerte el nombre de un libro, aunque éste esté en *itálicas* en una pantalla. La mayoría del marcado presentacional ha sido desechada con HTML 4.0, en favor de hojas de estilo en cascada.

El marcado *hipertextual* se utiliza para enlazar partes del documento con otros documentos o con otras partes del mismo documento. Para crear un enlace es necesario utilizar la etiqueta de ancla `<a>` junto con el atributo `href`, que establecerá la dirección URL a la que apunta el enlace. Por ejemplo, un enlace a la Wikipedia sería de la forma `Wikipedia`. También se pueden crear enlaces sobre otros objetos, tales como imágenes ``.

1.1.7.3.3.2 Atributos

La mayoría de los atributos de un elemento son pares nombre-valor, separados por un signo de igual «=» y escritos en la etiqueta de comienzo de un elemento, después del nombre de éste. El valor puede estar rodeado por comillas dobles o simples, aunque ciertos tipos de valores pueden estar sin comillas en HTML (pero no en XHTML). De todas maneras, dejar los valores sin comillas es considerado poco seguro. En contraste con los pares nombre-elemento, hay algunos atributos que afectan al elemento simplemente por su presencia (tal como el atributo `ismap` para el elemento `img`).

1.1.7.3.4 Códigos HTML básicos

- `<html>`: define el inicio del documento HTML, le indica al navegador que lo que viene a continuación debe ser interpretado como código HTML. Esto es así de facto, ya que en teoría lo que define el tipo de documento es el DOCTYPE, que significa la palabra justo tras DOCTYPE el tag de raíz, por ejemplo:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Strict//EN"
"http://www.w3.org/TR/html4/strict.dtd">
```

- `<script>`: incrusta un script en una web, o llama a uno mediante `src="url del script"`. Se recomienda incluir el tipo MIME en el atributo `type`, en el caso de JavaScript `text/javascript`.
- `<head>`: define la cabecera del documento HTML; esta cabecera suele contener información sobre el documento que no se muestra directamente al usuario como, por ejemplo, el título de la ventana del navegador. Dentro de la cabecera `<head>` es posible encontrar:

Un ejemplo de código HTML con coloreado de sintaxis.

- `<title>`: define el título de la página. Por lo general, el título aparece en la barra de título encima de la ventana.
- `<link>`: para vincular el sitio a hojas de estilo o iconos. Por ejemplo: `<link rel="stylesheet" href="/style.css" type="text/css">`.
- `<style>`: para colocar el estilo interno de la página; ya sea usando CSS u otros lenguajes similares. No es necesario colocarlo si se va a vincular a un archivo externo usando la etiqueta `<link>`.
- `<meta>`: para metadatos como la autoría o la licencia, incluso para indicar parámetros http (mediante `http-equiv=""`) cuando no se pueden modificar por no estar disponible la configuración o por dificultades con server-side scripting.
- `<body>`: define el contenido principal o cuerpo del documento. Esta es la parte del documento html que se muestra en el navegador; dentro de esta etiqueta pueden definirse propiedades comunes a toda la página, como color de fondo y márgenes. Dentro del cuerpo `<body>` es posible encontrar numerosas etiquetas. A continuación se indican algunas a modo de ejemplo:

`<h1>` a `<h6>`: encabezados o títulos del documento con diferente relevancia.

- `<table>`: define una tabla.
- `<tr>`: fila de una tabla.

- `<td>`: celda de una tabla (debe estar dentro de una fila).
- `<a>`: hipervínculo o enlace, dentro o fuera del sitio web. Debe definirse el parámetro de pasada por medio del atributo `href`. Por ejemplo: `Wikipedia` se representa como Wikipedia).
- `<div>`: división de la página. Se recomienda, junto con `css`, en vez de `<table>` cuando se desea alinear contenido.
- ``: imagen. Requiere del atributo `src`, que indica la ruta en la que se encuentra la imagen. Por ejemplo: ``. Es conveniente, por accesibilidad, poner un atributo `alt="texto alternativo"`.
- ``: etiquetas para listas.
- ``: texto en negrita (etiqueta desaprobadada. Se recomienda usar la etiqueta ``).
- `<i>`: texto en cursiva (etiqueta desaprobadada. Se recomienda usar la etiqueta ``).
- `<s>`: texto tachado (etiqueta desaprobadada. Se recomienda usar la etiqueta ``).
- `<u>`: Antes texto subrayado. A partir de HTML 5 define porciones de texto diferenciadas o destacadas del resto, para indicar correcciones por ejemplo. (etiqueta desaprobadada en HTML 4.01 y redefinida en HTML 5)

La mayoría de etiquetas deben cerrarse como se abren, pero con una barra (`</>`) tal como se muestra en los siguientes ejemplos:

```
<table><tr><td>Contenido de una celda</td></tr></table>.
```

```
<script>Código de un [[script]] integrado en la página</script>.
```

El lenguaje HTML puede ser creado y editado con cualquier editor de textos básico, como puede ser Gedit en Linux, el Bloc de notas de Windows, o cualquier otro editor que admita texto sin formato como GNU Emacs, Microsoft Wordpad, TextPad, Vim, Notepad++, entre otros.

Existen, además, otros editores para la realización de sitios web con características WYSIWYG (What You See Is What You Get, o en español: «lo que ves es lo que obtienes»). Estos editores permiten ver el resultado de lo que se está editando en tiempo real, a medida que se va desarrollando el documento. Ahora bien, esto no significa una manera distinta de realizar sitios web, sino que una forma un tanto más simple, ya que estos programas, además de tener la opción de trabajar con la vista preliminar, tiene su propia sección HTML, la cual va generando todo el código a medida que se va trabajando. Algunos ejemplos de editores WYSIWYG son KompoZer, Microsoft FrontPage o Adobe Dreamweaver.

Combinar estos dos métodos resulta muy interesante, ya que de alguna manera se ayudan entre sí. Por ejemplo, si se edita todo en HTML y de pronto se olvida algún código o etiqueta, simplemente me dirijo al editor visual o WYSIWYG y se continúa ahí la edición o viceversa, ya que hay casos en que resulta más rápido y fácil escribir directamente el código de alguna característica que el usuario desea adherir al sitio que buscar la opción en el programa mismo.

Existe otro tipo de editores HTML llamados WYSIWYM que dan más importancia al contenido y al significado que a la apariencia visual. Entre los objetivos que tienen estos editores es la separación del contenido y la presentación, fundamental en el diseño web.

HTML utiliza etiquetas o marcas, que consisten en breves instrucciones de comienzo y final, mediante las cuales se determinan la forma en la que debe aparecer en su navegador el texto, así como también las imágenes y los demás elementos, en la pantalla del ordenador.

Toda etiqueta se identifica porque está encerrada entre los signos menor que y mayor que (<>), y algunas tienen atributos que pueden tomar algún valor. En general las etiquetas se aplicarán de dos formas especiales:

Se abren y se cierran, como por ejemplo: negrita, que se vería en su navegador web como negrita.

No pueden abrirse y cerrarse, como `<hr />`, que se vería en su navegador web como una línea horizontal.

Otras que pueden abrirse y cerrarse, como por ejemplo `<p>`.

Las etiquetas básicas o mínimas son:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">
```

```
<html lang="es">
```

```
<head>
```

```
<title>Ejemplo1</title>
```

```
</head>
```

```
<body>
```

```
<p>ejemplo1</p>
```

```
</body>
```

```
</html>
```

1.1.7.3.5 Clase de Páginas WWW en el lenguaje HTML.

Detallamos a continuación los pasos a seguir para crear una página web, luego de esto se pueden ir ingresando los comandos que se necesiten para agregar: textos, imágenes y sonidos.

Primero se aconseja crear un directorio donde se van a guardar todos los archivos de la página web que se va a utilizar, esto se debe a que las imágenes, sonidos o videos no se guardan dentro del archivo sino en forma separada (no como en Word). En HTML las páginas Web sólo contienen texto, no están guardadas en ellas propiamente tal como ningún otro tipo de archivo.

Abrir un editor de texto básico del computador (notepad), para luego guardar un archivo con el código en HTML. La extensión con que se guarda el archivo tiene que

ser (.html o .htm), por ejemplo tarea.htm. Se pueden dar cuenta que los archivos de Word tienen la extensión .doc y los de excel .xls, pueden ver esto en las clases de Sistema Operativos que vimos con anterioridad. Ojo el archivo de inicio de las páginas debiera ser index.htm, porque es el archivo por defecto que ve un browser en un determinado directorio (Dirección).

Tener abierto un browser con el archivo que se está creando para ver los resultados inmediatamente, para así ir creando y diseñando la página web.

A continuación veremos varios de los códigos del lenguaje y sus usos para que el alumno pueda ver los resultados en forma inmediata.

```
<html>
```

```
</html>
```

Para indicarle al browser que es código HTML el que viene descrito a continuación, se limita todo el código por estos dos parámetros. Dentro de ellos se escribe el código de la página

```
<title>
```

```
</title>
```

Entre estos dos comandos va escrito el título de la página, este va en la parte superior de la ventana que despliega el browser.

```
<body bgcolor=# text=# link=# vlink=# alink=# background="">
```

```
</body>
```

Entre estos parámetros se escribe todo el texto de la página, el cuerpo del documento (BODY). Con estos comandos se define el color general de:

bgcolor=# color del fondo de la página, este color puede ser cambiado por un fondo, (background)

text=# es el color del texto general de la página

link=# color del texto con link

vlink=# color del texto de link visitado

alink=# color del texto de link activo

background="" en este comando se puede detallar un fondo específico para la página, un archivo de imagen.

Los colores se definen en formato RGB, se puede ver ejemplos de los colores con sus códigos en al siguiente página COLORES. Estos colores consisten de seis dígitos, o letras desde la A hasta la F.

Este comando sirve para insertar un salto de línea dentro de la página.

<p>

Sirve para ingresar un salto de párrafo dentro del contenido

<h!!>

</h!!>

Especifica el tipo de título con un número del 1 al 6 en vez de !!, especificando la importancia del título.

Ejemplo de Título 1

Sirve para especificar las fuente del texto que va entre estos dos parámetros.

face="Arial,Helvetica" Especifica el tipo de letra a utilizar.

color=# Especifica el color del texto a utilizar, descrito en anterioridad
Los colores se definen en formato RGB, se puede ver ejemplos de los colores con sus

códigos en al siguiente página COLORES. Estos colores consisten de seis dígitos, o letras desde la A hasta la F.

size=%% Detalla el tamaño de la letra elegido, número entre el -6 y el +6, en vez de %%.

Ejemplo de Font Arial, Helvetica y color #000000

El texto que va entre estos dos parámetros se encuentra en BOLD, o Negritas.

Ejemplo de Bold

<i></i>

El texto que va entre estos dos comandos se encuentra en Itálicas.

Ejemplo de Itálicas

<u></u>

El texto que va entre estos dos parámetros está subrayado.

Ejemplo de Bold

<imgsrc="\$\$">

Este comando inserta imágenes dentro de las páginas web, tiene que estaespecificado el directorio en el caso que no se encuentre la imagen en el mismo directorio que la página.

 Detalla el alineamiento que tendría la imagen (derecho=right, izquierdo=left y centrado=center)

<ahref="\$\$">

Sirve para realizar direccionamiento a nuevas páginas webs, LINKS. Puede ser un texto o una imagen. En vez de \$\$, se puede colocar la dirección a donde va a

apuntar, <http://www.ing.puc.cl> o puede ser [colores.htm](#), también puede ser reemplazada por <mailto:andres@ing.puc.cl> y así aparecerá una ventana de mail a andres@ing.puc.cl al apretar el texto o imágenes que se encuentren dentro de estos parámetros.

ñ

Con este comando se escriben la ñ.

Se utiliza para ingresar espacios dentro del texto, para ingresar más de un espacio en la página.

1.1.7.3.6 Tabulaciones

Necesarios para tabular alguna sección de la página, desde textos, imágenes y tablas.

Ejemplo de Tabulación

Ayuda a ingresar viñetas dentro del texto.

Ejemplo de Viñeta

...

Este comando inserta viñetas con números, estos son contiguos.

Ejemplo de Viñeta Numerada

<center></center>

Centra un objeto, texto, imagen, tabla, etc.

```
<align="left">
izquierda</p>
```

Deja el texto a la izquierda de la página

```
<align="right">
derecha </p>
```

Deja el texto a la derecha de la página

```
<table>
<tr>
<td colspan="N">
<td rowspan="N">
</td>
</tr>
```

Comandos para definir una tabla,

```
<tr></tr>
```

Se utiliza para definir una fila.

```
<td></td>
```

Define una columna de la tabla

Ej de una tabla de de 2 por 3, dos columnas por tres filas

```
<tableborder=1>
<tr> <td>1</td>
<td>2</td> <td>3</td>
</tr> <tr>
<td>4</td>
<td>5</td>
```

1	2	3
4	5	6

<td>6</td>

</tr></table>

Para generar una celda de varias filas o columnas, se logra poniendo un número en la posición de "N", siendo este desde 2 en adelante, el uno no tiene sentido, porque se coloca el comando <tr>.

1.1.7.3.7 El HTML (Hiptertext Make up Language)

Son comandos con los que se desarrollan las páginas WEB. A continuación hay una lista con algunos de los comandos usados en HTML.

INICIO		FINAL	COMENTARIOS
Obligatorias			
Principio de Documento	<html>	</html>	Toda la codificación debe estar incluida en estas etiquetas
Encabezado	<head>	</head>	Dentro del encabezado se encontrará el título.
Título	<title>	</title>	Es el texto que aparece en la barra de títulos del navegador
Cuerpo	<body>	</body>	Cuerpo de la página
Para toda la página			
Color de Fondo	<body bgcolor="#RRVVAA"> Ver escala de Colores		Permite colocar un color uniforme de fondo a toda la página. Este color estará formado por 3 números hexadecimales que indiquen: RR = tonalidad de color rojo VV = tonalidad del color verde AA = tonalidad del color azul
Imagen de Fondo	<body background="nombreimagen.gif">		nombreimagen.gif deberá reemplazarse por el nombre de la imagen. Se recomienda utilizar formatos .gif o .jpg
Colores de texto color de enlaces enlaces visitados	<body text="#RRVVAA" Link="#RRVVAA" Vlink="#RRVVAA" Alink="#RRVVAA">		Permite definir los colores que se aplicarán en forma genérica para el texto como para los enlaces. El color está representado por un número hexadecimal

enlaces activos	Ver escala de Colores		
Estilos	<style>	</style>	Permite definir un estilo de fuente, color, tamaño, etc para todo el documento.
Meta	<meta name="description" content="comentarios" name="keywords" content="palabra1 palabra2 ... palabran">	</meta>	Permite definir propiedades internas del documento. Name = "description" content = "Será el resumen con que se publicará en el buscador. No utilizar más de 25 palabras" Name = "keywords" content = "podrán definirse palabras claves para que nuestro documento sea encontrado por los buscadores"
Sonido	<bgsound src="xx.wav" loop=infinite/n> <embed src="xx.wav" width=200 height=55 autostart="true" loop="true" hidden="true">		Ejecuta un archivo de sonido. xx.wav se debe reemplazar por el nombre del archivo; loop n indica la cantidad de veces que se repetirá. La etiqueta Embed con sus propiedades, se utiliza por las incompatibilidades de los navegadores. Con esta opción aparece una consola cuyas dimensiones se definen con Width o Height y que puede ocultarse con Hidden = "true"
Presentación de texto			
Encabezamientos	<h1><h2>....<h6>	</h1></h2>....</h6>	Opciones de formato de texto para encabezados,
Negrita			Coloca el texto incluido con éstos formatos.
Cursiva	<i>	</i>	
Subrayado	<u>	</u>	
Parpadeo	<blink>	</blink>	El texto estará parpadeante
Grande	<big>	</big>	Agranda el texto
Pequeña	<small>	</small>	Disminuye el texto
Subíndice	_		Representa el texto sobre el renglón o bajo el renglón.
Superíndice	[]	
Color del Texto			Permite colocar un color a la fuente. El color está representado por un número hexadecimal.
Tamaño del Texto			Define un tamaño de fuente específico. n representa un número del 0 al 7 al que luego podrán agregarle los signos + o -

Tipo de fuente			Permite definir un nombre de Fuente específico.																																										
Estilo de fuente	<tt>	</tt>	Formato de fuente Courier de tamaño menor (Typewriter)																																										
Texto preformateado	<pre>	</pre>	Formato de fuente tipo Courier. Se representan los espacios en blanco																																										
Texto en Movimiento	<marquee>(texto)	</marquee>	Permite que un texto tenga movimiento. Pueden agregarse atributos para dimensionar la marquesina, para alinear el texto, para modificar tamaño, fuente y color.																																										
Línea Horizontal	<hr="n">	No utiliza etiqueta de cierre	Traza un línea horizontal cuyo grosor está representado por "n"																																										
Espacio en Blanco	 	No utiliza etiqueta de cierre	Representa un espacio en blanco.																																										
Comentario	<!-- comentario>	->	Comentarios que no serán visibles en la pantalla.																																										
Caracteres especiales	<table border="1"> <tr> <td>Escribo</td> <td>Para ver</td> </tr> <tr> <td>&lt;</td> <td><</td> </tr> <tr> <td>&gt;</td> <td>></td> </tr> <tr> <td>&amp;</td> <td>&</td> </tr> <tr> <td>&quot;</td> <td>"</td> </tr> <tr> <td>&aacute;</td> <td>á</td> </tr> <tr> <td>&eacute;</td> <td>é</td> </tr> <tr> <td>&iacute;</td> <td>í</td> </tr> <tr> <td>&oacute;</td> <td>ó</td> </tr> <tr> <td>&uacute;</td> <td>ú</td> </tr> <tr> <td>&Aacute;</td> <td>Á</td> </tr> <tr> <td>&Eacute;</td> <td>É</td> </tr> <tr> <td>&Iacute;</td> <td>Í</td> </tr> <tr> <td>&Oacute;</td> <td>Ó</td> </tr> <tr> <td>&Uacute;</td> <td>Ú</td> </tr> <tr> <td>&ntilde;</td> <td>ñ</td> </tr> <tr> <td>&Ntilde;</td> <td>Ñ</td> </tr> <tr> <td>&uuml;</td> <td>ü</td> </tr> <tr> <td>&Uuml;</td> <td>Ü</td> </tr> <tr> <td>&#191;</td> <td>¿</td> </tr> <tr> <td>&#161;</td> <td>¡</td> </tr> </table>	Escribo	Para ver	<	<	>	>	&	&	"	"	á	á	é	é	í	í	ó	ó	ú	ú	Á	Á	É	É	Í	Í	Ó	Ó	Ú	Ú	ñ	ñ	Ñ	Ñ	ü	ü	Ü	Ü	¿	¿	¡	¡		<p>Permiten que los caracteres especiales sean leídos por todos los navegadores en sus distintas versiones.</p> <p>La escritura de cada carácter comienza con el signo ampersand (&) y debe terminar con punto y coma (;)</p> <p>En esta tabla vemos como escribir algunos símbolos, signos, las letras Ñ y ñ, y las vocales acentuadas en mayúscula y minúscula.</p>
Escribo	Para ver																																												
<	<																																												
>	>																																												
&	&																																												
"	"																																												
á	á																																												
é	é																																												
í	í																																												
ó	ó																																												
ú	ú																																												
Á	Á																																												
É	É																																												
Í	Í																																												
Ó	Ó																																												
Ú	Ú																																												
ñ	ñ																																												
Ñ	Ñ																																												
ü	ü																																												
Ü	Ü																																												
¿	¿																																												
¡	¡																																												

Presentación de párrafos			
Alineaciones	<center> <left> <right>	</center> </left> </right>	Todo texto que se escriba entre estas etiquetas sufrirá esa alineación .
Sangría	<blockquote>	</blockquote>	Se utiliza para destacar una cita.
Párrafo	<p align= center*left*right*justify>	</p>	Realiza la separación entre párrafos, y la alineación de estos.
Renglones en blanco	 		Permite dejar renglones en blanco
Tratamiento de imágenes			
Imagen Individual			Permite insertar una imagen en la página que estamos diseñando. El archivo de la imagen deberá estar ubicado en la misma carpeta que la página, caso contrario se deberá indicar su ruta de acceso
Texto de la imagen			Coloca un texto que podrá leerse al pasar el puntero del ratón sobre la imagen o cuando ésta no se carga.
Alineación del texto			Indicará la posición en que ubicaremos al texto que acompaña a la imagen Top Arriba * Middle Medio * Bottom Abajo
Ancho Alto			Define el tamaño de la imagen Width = ancho * height = altura n = será un valor en pixeles
Videos			Permite agregar archivos de video.
Listas			
Lista numerada	<ol type= A * a * I * i start=n> <lh> título de la lista </lh> Primera opción Segunda opción Tercera opción		Se utiliza cuando las opciones deben ser numeradas. La opción type representan número o letras y start indicará el número con que inicia.
Lista no ordenada	<ul type=square * circle * disk> <lh> título de la lista </lh> Primera opción Segunda opción		Se utiliza cuando las opciones no presentan un orden determinado. La opción type representa la viñeta.

	 Tercera opción		
Lista con sangrado	<dl> <lh> título de la lista </lh> <dt> Primer tema <dd> Primer detalle <dd> Segundo detalle <dt> Segundo tema <dd> Tercer detalle <dd> Cuarto detalle	</dl>	Se utiliza cuando las opciones llevan un título y una definición
Enlaces o Links			
Enlace a otro URL	 yyy		xxx se debe reemplazar por la dirección URL destino del enlace. yyy es el texto indicativo que se leerá en la página.
Enlace a un e-mail	 texto indicativo del enlace		Te permite un enlace directo a tu cliente de correo predeterminado
Marca para enlace dentro de la misma página (Ancla - Anchor)	 Texto de enlace al ancla		Referencia una posición a la que luego se accederá.
Localizar enlace anterior	 texto del ancla		Enlaza una posición previamente marcada, dentro de la misma página

Tabla Nº 1 Comandos usados en HTML

1.1.7.3.8 Estructura de una Página

Bueno, esto ya se empieza a poner interesante. Vamos a empezar a escribir nuestro primer código HTML. Lo primero que hay que saber es que todo documento HTML debe empezar y acabar por las etiquetas <HTML> y </HTML> respectivamente. Como veis empezamos a utilizar este lenguaje. Todas las etiquetas van entre <>. Además la gran mayoría de las etiquetas van emparejadas y para distinguir entre la etiqueta de inicio y de fin, se le pone / a la etiqueta de fin. Fácil no? Todos los documentos HTML tienen dos partes, el encabezado y el cuerpo, de Perogrullo no? Pues hasta para esto hay que poner etiqueta, y cuales son? Para el encabezado la etiqueta es <HEAD> y </HEAD> En esta sección se pone la

información de la página y poco más. Aquí es donde ponemos el título de nuestra página, para que todo el mundo nos conozca. El título irá entre las etiquetas <TITLE> y </TITLE> lo que pongamos aquí, aparecerá en la parte izquierda de la línea superior de la ventana.

Ya esta, manos a la obra vamos a empezar a diseñar nuestra primera página. Cogemos cualquier procesador, y acordaros de que se debe salvar como texto normal.

```
<HEAD>  
<TITLE>MipaginaWeb</TITLE>  
  
</HEAD>
```

❖ Párrafos

Haber visto las posibilidades de alineamiento del párrafo, pues hay más. Si lo que queremos es sangrar un párrafo, debemos escribir entre las etiquetas <BLOCKQUOTE> Y </BLOCKQUOTE>.

Los párrafos van entre las etiquetas <P> y </P>, y los podemos alinear a la izquierda, a la derecha y como ya hemos visto, centrarlos, pero vamos a seguir con nuestra página, introduciendo el comando ALIGN.

```
<BODY>  
<CENTER><H3>BIENVENIDO A MI PAGINA PERSONAL</H3></CENTER>  
<P ALIGN="LEFT"> Ahora estoy escribiendo el primer párrafo de mi página personal,  
que quiero que se alinea a la izquierda. </P>  
<P ALIGN="RIGHT"> Pero esta línea quiero que salga a la derecha</P>  
<P ALIGN="CENTER"> Toma ya y ahora en el centro</P>  
</BODY>
```

Bienvenido A Mi Pagina Personal (H2)

Los párrafos van entre las etiquetas <P> y </P>, y los podemos alinear a la izquierda, a la derecha y como ya hemos visto, centrarlos, pero vamos a seguir con nuestra página, introduciendo el comando ALIGN.

```
<BODY>
<CENTER><H3>BIENVENIDO A MI PAGINA PERSONAL</H3></CENTER>
<P ALIGN="LEFT"> Ahora estoy escribiendo el primer párrafo de mi página personal,
que quiero que se alinee a la izquierda. </P>
<P ALIGN="RIGHT"> Pero esta línea quiero que salga a la derecha</P>
<P ALIGN="CENTER"> Toma ya y ahora en el centro</P>
</BODY>
```

Ahora estoy escribiendo el primer párrafo de mi página personal, que quiero que se alinee a la izquierda
Pero esta línea quiero que salga a la derecha
Toma ya y ahora en el centro.

❖ Formatos

Pues visto lo anterior, seguimos dando forma a nuestra página, ahora vamos a ver como darle formato a nuestro texto. Lo primero que vamos a ver es como modificar el tamaño de la letra, esto se hace con el comando , en unión con el modificador SIZE.

Hay varias maneras de modificar el tamaño de la fuente. En primer lugar podemos hacer referencia al tamaño absoluto de la fuente. Los tamaños absolutos varían entre el 1 y el 7, siendo 7 el más grande, y 1 el más pequeño. La segunda manera de realizar la modificación, es hacer referencia al tamaño relativo de la fuente, es decir restando o incrementando el tamaño actual de la fuente en la cantidad especificada. Este modificador puede variar entre -4 y +4. Pero como mejor se ve esto es continuando con nuestra página.

```
<P><FONT SIZE="3">Esta línea está escrita con tamaño 3</FONT></P>
```

Esta línea está escrita con tamaño 3

```
<P><FONT SIZE="5">Esta línea está escrita con tamaño 5</FONT></P>
```

5</P>

Esta línea esta escrita con tamaño 5

<P>Esta línea está escrita con tamaño -

3</P>

Esta línea está escrita con tamaño relativo -3

Cómo vemos en el último ejemplo, el resultado es el mismo que si aplicamos , resultado que restarle 3 unidades al último tamaño definido de fuente, en nuestro ejemplo 5.

Visto todo lo anterior, sólo queda por decir que es posible determinar cuál es el tamaño base de la fuente con la que vamos a escribir nuestro documento HTML, para ello usamos la etiqueta <BASEFONT SIZE=tamaño>, escrito justo a continuación de la etiqueta <BODY>.

❖ Estilos

Para resaltar partes del texto que estamos escribiendo, podemos utilizar negritas, cursivas, etc. Las etiquetas más comunes que podemos usar aparecen en la siguiente

Tabla.

Estilo	Etiqueta	
Negrita		Negrita
Cursiva	<I></I>	<i>Cursiva</i>
Subrayado	<U></U>	<u>Subrayado</u>
Cita	<CITE></CITE>	<i>Cita</i>
Código	<CODE></CODE>	Código
Enfasis		<i>Énfasis</i>
Fuerte		Fuerte
Teletipo	<TT></TT>	Teletipo
Tachado	<STRIKE></STRIKE>	Tachado
Subíndice		Subíndice
Superíndice		Superíndice
Grande	<BIG></BIG>	Grande
Pequeño	<SMALL></SMALL>	Pequeño

Tabla: Nº 2 Etiqueta en HTML

Esto se va a escribir en negrita</P>

❖ Líneas Horizontales

A veces vamos a necesitar separar las distintas secciones de nuestro documento HTML, una forma sencilla de hacerlo, es insertar líneas horizontales. Esto se consigue con la etiqueta <HR>. Esta etiqueta no lleva su pareja de cierre, es individual, por defecto traza una línea de todo el ancho de la pantalla, no obstante se puede modificar el aspecto de las líneas añadiendo los siguientes parámetros:

SIZE = Indica el grosor en pixel de la línea

WIDTH = Especifica la anchura que la línea ocupa en la pantalla, se le puede especificar en pixeles o en porcentaje.

ALIGN= Alineación de la línea

NOSHADE = Si queremos la línea con sombra o sin sombra.

Pero todo esto se ve mejor con un ejemplo:

```
<HR SIZE="5" WIDTH="50%" ALIGN="LEFT" NOSHADE="NOSHADE">
```

El resultado es una línea de grosor 5 pixel un 50% de la pantalla alineada a la izquierda sin sombra, es decir:

```
<P><B>Esto se va a escribir en negrita</B></P>
```

❖ Enlaces

La verdadera potencia de internet nos la da la posibilidad de enlazar todas las páginas como si de una tela de araña se tratase, una página sin enlaces no tiene demasiado sentido, así que vamos a ver de forma muy breve como insertar enlaces dentro de nuestro documento. Los enlaces nos pueden llevar a otro documento o bien a otra sección dentro del documento editado. Este documento de tutorial, está repleto de enlaces a la misma página, como haber podido ver, desde el índice que está al principio de este documento, se accede a las secciones directamente, igualmente de cada una de las secciones, se accede al índice. Además, este documento tiene enlaces externos, como veras en la parte inferior de la pantalla, hay un enlace a la página principal de mi sita y hay otro enlace a

Por cierto si os gusta leer y escuchar música, pinchar en este enlace, poder adquirir los mejores libros y la mejor música de internet, y además así colaborareis a que pueda seguir manteniendo este tutorial.

Bueno, y ahora manos a la obra que es lo que nos interesa, vamos a generarnos un enlace a uno de los buscadores mas conocidos YAHOO

```
<A HREF="http://YAHOO.COM">Buscar en YAHOO</A>
```

Buscar en YAHOO

Ahora lo vamos a complicar un poco más y vamos a poner un gráfico como el del ejemplo anterior.

```
<PALIGN="CENTER"><AHREF="http://amazon.com"><IMG  
SRC="amazon.gif"></A></P>
```

Bien no, es mucho más fácil de lo que parece. Como veis Hemos realizado dos enlaces a otras páginas pero con el mismo formato podemos acceder a otros muchos servicios de internet, podemos realizar enlace tipo ftp, correo electrónico, etc. Vamos a ver como añadir un enlace de correo:

```
<A HREF="mailto:f_casa@yahoo.com">Mándame un correo</A>
```

Mándame un correo

Para agregar enlaces a marcadores dentro de la página actual, o dentro de otra página, lo primero que hay que definir es el marcador, esto se hace de la siguiente manera:

```
<A NAME="Nombre del marcador">Texto del marcador </A>
```

Texto del marcador

El enlace al marcador anterior se definiría:

```
<A HREF="#Nombre del marcador">Enlace al marcador anterior</A>
```

Cómo véis se incluye el símbolo "#" antes del nombre del marcador También se pueden hacer llamadas a marcadores definidos en otras páginas:

`Enlace al marcador de la página
index.html`

❖ Dar Color a la Página

Hasta ahora hemos visto todo lo necesario para realizar nuestra primera página WEB, pero ahora, vamos a darle un poco de vidilla, ya que una página con letra negra sobre fondo blanco, queda un poco sosa, así que vamos a modificar los colores tanto del fondo como de las letras, e incluso vamos a poder insertar un gráfico como fondo de nuestra página.

Para modificar los colores de fondo del documento debemos añadir a la etiqueta de cuerpo de documento `<BODY>` el parámetro `BGCOLOR="#RRGGBB"` donde `#RRGGBB` indica la combinación de tonos de rojo, verde y azul necesarios para conseguir el color que deseamos. No obstante HTML reconoce 16 colores simples black, gray, maroon, purple, green, olive, navy, teal, silver, red, white, lime, yellow, aqua, blue y fuchsia con nombre y apellidos, así que para poner un fondo azul, bastará con poner `<BODY BGCOLOR="BLUE">` sin más. Si lo queremos complicar más, necesitamos saber el número en hexadecimal que define el color que queremos y lo ponemos en la etiqueta `bgcolor`, el color rojo sin ir mas lejos sería `#FF0000`. Para definir el color del texto, utilizaremos el parámetro `TEXT="#RRGGBB"`. Para definir el color de los enlaces no visitados, `LINK="#RRGGBB"`, para los enlaces visitados `VLINK="#RRGGBB"`, para los enlaces en los que se haga click `ALINK="#RRGGBB"`.

Para insertar una imagen de fondo, usamos el parámetro `BACKGROUND="imagenfondo.gif"`. Los formatos de ficheros más utilizados son el gif y jpg. Cabe reseñar que el fichero debe estar en el mismo directorio que el documento HTML que lo llama, en caso contrario se debe especificar el `PATH` a dicho fichero gráfico. Y OJO, ya dije anteriormente que aunque en el código HTML da igual escribir las etiquetas en mayúsculas o minúsculas, cuando se hace una llamada a un fichero,

enlace, etc. Si existen diferencias, con lo cual para evitar problemas, escribimos todo en minúsculas.

❖ Insertar Imágenes

Y como todo documento que se precie debe tener imágenes, vamos a insertarlas de una manera facilísima, esto se hace con el comando ``. Si además queremos especificar el tamaño `` siendo n el tamaño en pixeles de la imagen. Si a esto le añadimos todo lo que hemos visto anteriormente, podemos alinear la imagen, incluirla en un enlace a otras páginas, etc.
`<PALIGN="CENTER"><AHREF="http://armazon.com"></P>`

❖ Listas

Este es uno de los elementos más útiles para nuestros documentos HTML, ya que nos permite esquematizar de una forma clara la información contenida en nuestro documento.

Una lista es una sucesión de elementos que pertenecen a una estructura común y que es posible numerarlo, sin darle muchas más vueltas, el índice de esta página esta realizado con una lista.

Hay diferentes tipos de listas, sin orden con orden, numeradas, alfabéticas, etc.... Es bastante común que vayan precedidos de bolos o números para diferenciar una línea de otra.

Para crear una lista ordenada, se usan las etiquetas `` y ``, y a cada elemento se le incluye entre las etiquetas `<L1>` y `</L1>` pe.

``

INTRODUCCION

QUE ES HTML?

1.1.7.3.3 P H P.

1.1.7.3.3.1 Introducción

“PHP es un lenguaje de programación soportado por HTML usado generalmente para la creación de contenido para sitios web. PHP es un acrónimo recurrente o recursivo que significa "PHP Hypertext Pre-processor", y se trata de un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios Web.”

1.1.7.3.3.2 ¿Qué es PHP?

PHP es un lenguaje de programación, diseñado originalmente para la creación de páginas web dinámicas (con conexión a base de dato).

Una respuesta corta y concisa, pero que significa realmente. Un ejemplo nos aclarará las cosas:

Un ejemplo introductorio

```
<html>
```

```
<head>
```

```
<title>Example</title>
```

```
</head>
```

```
<body>
```

```
<?php
```

```
echo "Hi, I'm a PHP script!";
```

```
?>
```

```
</body>
```

```
</html>
```

Podemos ver que no es lo mismo que un script escrito en otro lenguaje de programación.

En vez de escribir un programa con muchos comandos para crear una salida en HTML, escribimos el código HTML con cierto código PHP embebido (introducido) en el mismo, que producirá cierta salida(en nuestro ejemplo, producir un texto). El código PHP se incluye entre etiquetas especiales de comienzo y final que nos permitirán entrar y salir del modo PHP.

Lo que distingue a PHP de la tecnología Javascript, la cual se ejecuta en la máquina cliente, es que el código PHP es ejecutado en el servidor. Si tuviésemos un script similar al de nuestro ejemplo en nuestro servidor, el cliente solamente recibiría el resultado de su ejecución en el servidor, sin ninguna posibilidad de determinar que código ha producido el resultado recibido. El servidor web puede ser incluso configurado para que procese todos los ficheros HTML con PHP.

Lo mejor de usar PHP es que es extremadamente simple para el principiante, pero a su vez, ofrece muchas características avanzadas para los programadores profesionales. No tengáis miedo de leer la larga lista de características de PHP, en poco tiempo podréis empezar a escribir vuestros primeros scripts.

Aunque el desarrollo de PHP está concentrado en la programación de scripts en la parte del servidor, se puede utilizar para muchas otras cosas. Sigue leyendo y descubre más sobre PHP en la sección ¿Qué se puede hacer con PHP?

1.1.7.3.3 Historia

PHP comenzó y sigue siendo primeramente usado como un lenguaje de script del lado del servidor embebido en HTML.

PHP, se conoce originalmente como Personal Home Pages, fue concebido en el otoño de 1994 por Rasmus Lerdorf. Él lo escribió como una forma de track visitantes a su CV en línea. La primera versión salió en los comienzos de 1995, y fue ahí donde Rasmus se dio cuenta que haciendo en proyecto código abierto, las personas arreglarían sus problemas. La primera versión fue muy precaria y tenía un parser que reconocía solo unas pocas macros y brindaba algunas utilidades que se usaban comúnmente en sitios web.

El parser fue reescrito a mediados de 1995 y se lo renombró a PHP/FI versión 2. El "FI" en esta versión quería decir Interprete formal. Lo que Rasmus había agregado a PHP fue de acuerdo a las necesidades crecientes de los sitios web. El soporte para MYSQL fue agregado. PHP/FI tuvo un crecimiento masivo, y otra gente empezó a contribuir programando regularmente.

A mediados de 1997 Zeev Suraski y Andi Gutmans reescribieron el parser principal, y PHP cambio de estar en manos de Rasmus a un grupo mas orientado al proyecto. Esto formo las bases para que PHP3, fuere ahora llamado PHP: Hypertext Preprocessor un acrónimo recursivo.

La última versión, de PHP4, es otra reescritura de Suraski and Gutmans y está basada en el motor Zend. PHP ahora tiene doscientos contribuyentes regularmente trabajando en varias partes del proyecto. Tiene una cantidad muy grande extensiones, módulos y soporta todos los servidores más populares nativamente, y además tiene soporte para MySql y ODBC.

Las últimas estadísticas muestran que PHP es actualmente usado por más de 5.5 millones de dominios, y ha tenido un gran crecimiento durante el último año. Es lejos el módulo más popular de Apache; para dar alguna perspectiva, Apache actualmente tiene un 60% del mercado de servidores de internet, y el servidor IIS (con soporte nativo para ASP) tiene menos de la mitad de esa proporción del mercado

1.1.7.3.3.4 ¿Qué se puede hacer con PHP?

PHP puede hacer cualquier cosa que se pueda hacer con un script CGI, como procesar la información de formularios, generar páginas con contenidos dinámicos. Y esto no es todo, se puede hacer mucho más.

Existen tres campos en los que se usan scripts escritos en PHP.

Scripts del lado del servidor. Este es el campo más tradicional y el principal foco de trabajo. Se necesitan tres cosas para que esto funcione. El intérprete PHP (CGI ó módulo), un servidor web y un navegador. Es necesario correr el servidor web con PHP instalado. El resultado del programa PHP se puede obtener a través del navegador, conectándose con el servidor web.

Scripts en la línea de comandos. Puede crear un script PHP y correrlo sin ningún servidor web o navegador. Solamente necesita el intérprete PHP para usarlo de esta manera. Este tipo de uso es ideal para scripts ejecutados regularmente desde cron (en Unix o Linux) o el Planificador de tareas (en Windows). Estos scripts también pueden ser usados para tareas simples de procesamiento de texto.

Escribir aplicaciones de interfaz gráfica. Probablemente PHP no sea el lenguaje más apropiado para escribir aplicaciones gráficas, pero si conoce bien PHP, y quisiera utilizar algunas características avanzadas en programas clientes, puede utilizar PHP-GTK para escribir dichos programas.

PHP puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes Unix, Microsoft Windows, Mac OS X, RISC OS y probablemente alguno más. PHP soporta la mayoría de servidores web de hoy en día, incluyendo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape y muchos otros. PHP tiene módulos disponibles para la mayoría de los servidores, para aquellos otros que soporten el estándar CGI, PHP puede usarse como procesador CGI.

De modo que, con PHP tiene la libertad de elegir el sistema operativo y el servidor de su gusto. También tiene la posibilidad de usar programación procedimental o programación orientada a objetos. Aunque no todas las características estándar de la programación orientada a objetos están implementadas en la versión actual de PHP, muchas bibliotecas y aplicaciones grandes (incluyendo la biblioteca PEAR) están escritas íntegramente usando programación orientada a objetos.

Con PHP no se encuentra limitado a resultados en HTML. Entre las habilidades de PHP se incluyen: creación de imágenes, archivos PDF y películas Flash (usando libswf y Ming) sobre la marcha. También puede presentar otros resultados, como XHTML y archivos XML. PHP puede auto generar estos archivos y almacenarlos en el sistema de archivos en vez de presentarlos en la pantalla.

Quizás la característica más potente y destacable de PHP es su soporte para una gran cantidad de bases de datos. Escribir un interfaz vía web para una base de datos es una tarea simple con PHP.

PHP puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes Unix (incluido HP-UX, Solaris y OpenBSD), Microsoft Windows, Mac OS X, RISC OS y probablemente alguno más. PHP soporta la mayoría de servidores web de hoy en día, incluyendo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape y iPlanet,

Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd y muchos otros. PHP tiene módulos disponibles para la mayoría de los servidores, para aquellos otros que soporten el estándar CGI, PHP puede usarse como procesador CGI.

Así que, con PHP tiene la libertad de escoger el sistema operativo y el servidor de vuestro gusto.

También tenía la posibilidad de usar programación de procedimientos ó programación orientada a objetos. Aunque no todas las características estándares de la programación orientada a objetos están implementadas en la versión actual de PHP,

muchas librerías y aplicaciones grandes (incluyendo la librería PEAR) están escritas íntegramente usando programación orientada a objetos.

Con PHP no está limitado a resultados en HTML. Entre las habilidades de PHP se incluyen, creación de imágenes, ficheros PDF y películas Flash (usando libswf y Ming) sobre la marcha. También puede presentar otros resultados, como XHTML y ficheros XML. PHP puede autogenerar estos ficheros y grabarlos en el sistema de ficheros en vez de presentarlos en la pantalla.

Quizás la característica más potente y destacable de PHP es su soporte para una gran cantidad de bases de datos. Escribir un interfaz vía web para una base de datos es una tarea simple con PHP. Las siguientes bases de datos están soportadas actualmente:

Adabas D	Ingres Oracle	(OCI7 and OCI8)
dBase	InterBase	Ovrimos
Empress	FrontBase	PostgreSQL
FilePro (read-only)	mSQL	Solid
Hyperwave Direct	MS-SQL	Sybase
IBM DB2	MySQL	Velocis
Informix	ODBC	Unix dbm

- **Lenguaje PHP Bases de la Sintaxis**

<? ?> Sólo si se activa la función short_tags() o la bandera de configuración short_open_tag.

- <?php ?>
- <script lenguaje="php"> </script>
- Sólo si se activan los tags para ficheros 'asp' con la bandera de configuración asp_tags.

- **Separación de Instrucciones**

Las instrucciones se separan con ';', en el caso de ser la última instrucción no es necesario el punto y coma.

Comentarios

Los comentarios en PHP pueden ser

- ✓ Como en C o C++, /*...*/ ó //
- ✓ Otro tipo de comentario de una línea es #, que comentará la línea en la que aparezca pero sólo hasta el tag ?> que cierra el código PHP.

- **Cookies HTML**

La función SetCookie() es una función PHP para asignar Cookies a un ordenador cliente. Esta función se debe llamar siempre antes de comenzar a crear la página puesto que debe formar parte de la cabecera de HTML. Cualquier Cookie que se envía a un cliente, se convierte dentro de PHP en una variable.

- **Cookies**

Para PHP las cookies son fáciles de utilizar. Para esto se utiliza la función setcookie() y al igual que la función anterior, Header, tiene que ser invocada antes de producir ninguna salida.

Esta función define la cookie que se va a enviar y posee todos los argumentos opcionales a excepción del nombre, aunque los no especificados deben ser indicados "", menos los parámetros expire que es un entero regular que puede devolver la función mktime(), y secure, que indica si el cookie debería ser utilizado sólo sobre conexiones seguras, enteros que deben ser sustituidos.

1.1.7 .3.2 Adobe Dreamweaver Cs3

1.1.7.3.2.1 Introducción

Dreamweaver CS3 es un software fácil de usar que permite crear páginas web profesionales.

Las funciones de edición visual de Dreamweaver CS3 permiten agregar rápidamente diseño y funcionalidad a las páginas, sin la necesidad de programar manualmente el código HTML.

Se puede crear tablas, editar marcos, trabajar con capas, insertar comportamientos JavaScript, etc., de una forma muy sencilla y visual.

Además incluye un software de cliente FTP completo, permitiendo entre otras cosas trabajar con mapas visuales de los sitios web, actualizando el sitio web en el servidor sin salir del programa.

1.1.7.3.2.2 Novedades de Dreamweaver CS3.

En este punto comentaremos las características que aporta esta nueva versión sobre la anterior:

- Mejoras en Spry. El "Framework" o estructura Spry ha recibido un fuerte impulso en esta versión de Dreamweaver. Spry es un conjunto de archivos JavaScript que permiten la inclusión de elementos dinámicos en las páginas HTML. Por ejemplo se pueden incluir acordeones, barras de menús, validación de formularios, acceso a datos XML, etc.
- También se han incluido efectos spry que se aplican a elementos ya existentes en la página HTML para hacer que se desvanezcan, reduzcan su tamaño, se resalten, etc.
- Mayor integración con Photoshop y Fireworks. Ahora es posible pegar directamente imágenes desde Photoshop en una página HTML de

Dreamweaver. Luego con un doble clic podremos ir a editar la imagen en Photoshop. Se pueden importar imágenes con capas.

- Mejoras en la comprobación de la compatibilidad con diferentes navegadores. Dreamweaver CS3 puede generar informes con los problemas de visualización de elementos CSS en los navegadores más utilizados.
- Adobe CSS Advisor es un sitio web accesible desde Dreamweaver que contiene información útil para resolver los problemas de compatibilidad entre navegadores.
- Mejoras en CSS. La administración de CSS es más fácil, ahora se pueden trasladar elementos CSS entre diferentes hojas de estilo. También ha aumentado la cantidad de plantillas CSS disponibles para aplicar a nuestras páginas web.
- Adobe Device Central, incluido en la Adobe Creative Suite, facilita la publicación en dispositivos móviles, PDAs, etc...

1.1.7.3.2.3 El Entorno de Trabajo.

Vamos a ver cuáles son los elementos básicos de Dreamweaver CS3, la pantalla, las barras, los paneles, etc. para saber diferenciar entre cada uno de ellos. Aprenderemos cómo se llaman, dónde están y para qué sirven. También veremos cómo obtener ayuda, por si en algún momento no sabemos cómo seguir trabajando. Cuando conozcamos todo esto estaremos en disposición de empezar a crear páginas web.

Al arrancar Dreamweaver aparece una pantalla inicial como ésta, vamos a ver sus componentes fundamentales. Así conoceremos los nombres de los diferentes elementos y será más fácil entender el resto del curso. La pantalla que se muestra a continuación (y en general todas las de este curso) puede no coincidir exactamente con la que ves en tu ordenador, ya que cada usuario puede decidir qué elementos quiere que se vean en cada momento, como veremos más adelante.

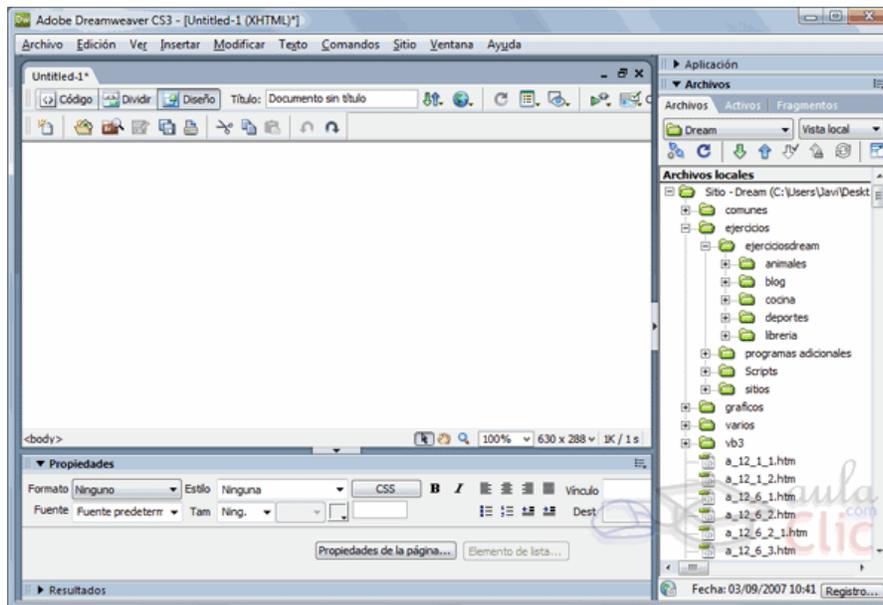


Grafico Nº 1: Pantalla de adobe dreamweaver

1. Configuración de un sitio local.

En este tema vamos a ver qué es un sitio web, cómo crearlos y gestionarlos, y cómo modificar las propiedades de los documentos, como puede ser el color de fondo.

2. Introducción.

Un sitio web es un conjunto de archivos y carpetas, relacionados entre sí, con un diseño similar o un objetivo común. Es necesario diseñar y planificar el sitio web antes de crear las páginas que va a contener.

La forma habitual de crear un sitio consiste en crear una carpeta en el disco local. Los documentos HTML normalmente se crean dentro de dicha carpeta, mientras que para contener las imágenes, las animaciones, las hojas de estilo, etc., se deben crear nuevas carpetas dentro de ésta, con el objetivo de tener una mejor organización de los archivos a la hora de trabajar. Esto es lo que se conoce como sitio local.

Después se podrán copiar los archivos en un servidor web, en el denominado sitio remoto, lo que equivale a publicar el sitio, de modo que la gente podrá verlo en Internet.

La organización de los archivos en un sitio permite administrar y compartir archivos, mantener los vínculos de forma automática, utilizar FTP para cargar el sitio local en el servidor, etc.

Es conveniente que la página inicial del sitio tenga el nombre index.htm o index.html, ya que los navegadores buscan una página con ese nombre cuando se intenta acceder a una URL genérica.

Por ejemplo, si escribiéramos la dirección genérica `http://www.aulaclic.com` en el navegador, éste intentaría cargar la página `http://www.aulaclic.com/index.htm`, por lo que se produciría un error en el caso de que no existiera ninguna página con el nombre index.htm.

Una vez creadas las carpetas que formarán un sitio local, ya es posible definir el sitio en Dreamweaver.

Para ello hay que dirigirse al menú Sitio, a la opción Administrar sitios.... Recuerda que a través del panel Archivos, pestaña Archivos, se puede acceder a cada uno de los sitios creados y a la opción Administrar sitio.

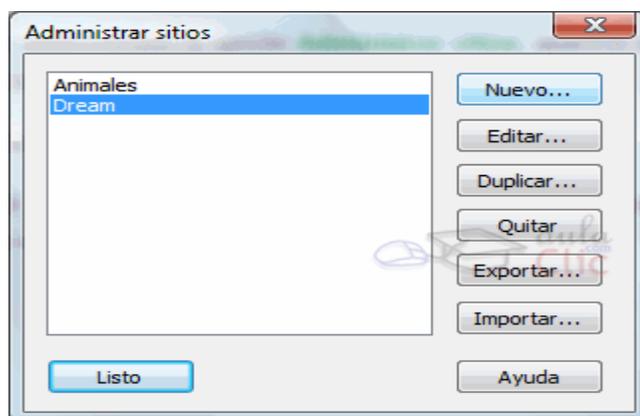


Grafico N° 2: Dream administrar sitios

En el caso de haber seleccionado la opción Administrar sitios, aparece una ventana que contiene la lista de sitios locales definidos con anterioridad. Por supuesto, pueden existir varios sitios locales en un mismo ordenador.

Tanto si se elige la opción Nuevo..., como si se elige la opción Editar..., se mostrará la misma ventana en la que definir las características del sitio.

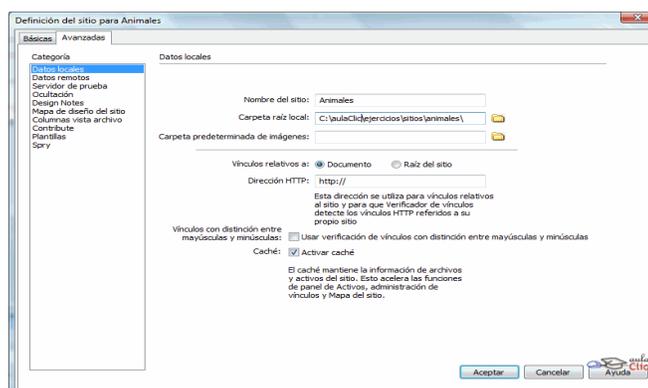


Grafico Nº 3: Pantalla Ubicación de la carpeta

Las características del sitio se agrupan en diferentes categorías que aparecen en la parte izquierda.

Para visualizar las características de una categoría basta con seleccionarla de la lista.

Haciendo clic en ella.

Vamos a ver los datos que hay que editar para la categoría Datos locales.

Debe definirse el Nombre del sitio y la Carpeta raíz local, que es en la que se encuentra el sitio dentro del disco duro local.

Después, si se desea, a través de la categoría Mapa de diseño del sitio puede definirse la página principal del sitio, de la que colgarán el resto de documentos HTML dentro del sitio, si en la carpeta raíz del sitio existe una página con el nombre index.htm, Dreamweaver la cogerá por defecto.

Estas tres características son las imprescindibles para definir un sitio local.

Las demás opciones en este momento no nos interesan, ya que estamos definiendo el sitio local, y no es necesario establecer los datos del servidor en el que estará el sitio remoto.

Después de rellenar los datos pulsamos el botón Aceptar y abrimos el sitio.

Si preferimos utilizar un asistente para crear el sitio web sólo tenemos que seleccionar la pestaña Básicas, en vez de la pestaña Avanzadas.

1.1.7 .3.3 Base de datos

Una base de datos o banco de datos (en ocasiones abreviada con la sigla *BD* o con la abreviatura *b. d.*) es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. Actualmente, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital (electrónico), que ofrece un amplio rango de soluciones al problema de almacenar datos.

Existen programas denominados sistemas gestores de bases de datos, abreviado SGBD, que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las propiedades de estos SGBD, así como su utilización y administración, se estudian dentro del ámbito de la informática.

Las aplicaciones más usuales son para la gestión de empresas e instituciones públicas. También son ampliamente utilizadas en entornos científicos con el objeto de almacenar la información experimental.

Aunque las bases de datos pueden contener muchos tipos de datos, algunos de ellos se encuentran protegidos por las leyes de varios países. Por ejemplo, en España los

datos personales se encuentran protegidos por la Ley Orgánica de Protección de Datos de Carácter Personal (LOPD).

1.1.7 .3.3.1 Tipos de base de datos

Las bases de datos pueden clasificarse de varias maneras, de acuerdo al contexto que se esté manejando, la utilidad de las mismas o las necesidades que satisfagan.

Según la variabilidad de los datos almacenados

- **Bases de datos estáticas**

Son bases de datos de sólo lectura, utilizadas primordialmente para almacenar datos históricos que posteriormente se pueden utilizar para estudiar el comportamiento de un conjunto de datos a través del tiempo, realizar proyecciones, tomar decisiones y realizar análisis de datos para inteligencia empresarial.

- **Bases de datos dinámicas**

Éstas son bases de datos donde la información almacenada se modifica con el tiempo, permitiendo operaciones como actualización, borrado y adición de datos, además de las operaciones fundamentales de consulta. Un ejemplo de esto puede ser la base de datos utilizada en un sistema de información de un supermercado, una farmacia, un videoclub o una empresa.

1.1.7 .3.4 Según el contenido:

- **Bases de datos bibliográficas**

Sólo contienen un subrogante (representante) de la fuente primaria, que permite localizarla. Un registro típico de una base de datos bibliográfica contiene información sobre el autor, fecha de publicación, editorial, título, edición, de una determinada publicación, etc. Puede contener un resumen o extracto de la publicación original, pero nunca el texto completo, porque si no, estaríamos en presencia de una base de datos a texto completo (o de fuentes primarias —ver más abajo). Como su nombre lo

indica, el contenido son cifras o números. Por ejemplo, una colección de resultados de análisis de laboratorio, entre otras.

- **Bases de datos de texto completo**

Almacenan las fuentes primarias, como por ejemplo, todo el contenido de todas las ediciones de una colección de revistas científicas.

- **Directorios**

Un ejemplo son las guías telefónicas en formato electrónico.

Bases de datos o "bibliotecas" de información química o biológica

Son bases de datos que almacenan diferentes tipos de información proveniente de la química, las ciencias de la vida o médicas. Se pueden considerar en varios subtipos:

Las que almacenan secuencias de nucleótidos o proteínas.

Las bases de datos de rutas metabólicas.

Bases de datos de estructura, comprende los registros de datos experimentales sobre estructuras 3D de biomoléculas.

Bases de datos clínicas.

Bases de datos bibliográficas (biológicas, químicas, médicas y de otros campos): PubChem, Medline, EBSCOhost.

1.1.7 .3.5 Modelos de bases de datos

Además de la clasificación por la función de las bases de datos, éstas también se pueden clasificar de acuerdo a su modelo de administración de datos.

Un modelo de datos es básicamente una "descripción" de algo conocido como *contenedor de datos* (algo en donde se guarda la información), así como de los

métodos para almacenar y recuperar información de esos contenedores. Los modelos de datos no son cosas físicas: son abstracciones que permiten la implementación de un sistema eficiente de *base de datos*; por lo general se refieren a algoritmos, y conceptos matemáticos.

Algunos modelos con frecuencia utilizados en las bases de datos:

- **Bases de datos jerárquicas**

En este modelo los datos se organizan en una forma similar a un árbol (visto al revés), en donde un *nodo padre* de información puede tener varios *hijos*. El nodo que no tiene padres es llamado *raíz*, y a los nodos que no tienen hijos se los conoce como *hojas*.

Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento.

Una de las principales limitaciones de este modelo es su incapacidad de representar eficientemente la redundancia de datos.

- **Base de datos de red**

Éste es un modelo ligeramente distinto del jerárquico; su diferencia fundamental es la modificación del concepto de *nodo*: se permite que un mismo nodo tenga varios padres (posibilidad no permitida en el modelo jerárquico).

Fue una gran mejora con respecto al modelo jerárquico, ya que ofrecía una solución eficiente al problema de redundancia de datos; pero, aun así, la dificultad que significa administrar la información en una base de datos de red ha significado que sea un modelo utilizado en su mayoría por programadores más que por usuarios finales.

- **Bases de datos transaccionales**

Son bases de datos cuyo único fin es el envío y recepción de datos a grandes velocidades, estas bases son muy poco comunes y están dirigidas por lo general al entorno de análisis de calidad, datos de producción e industrial, es importante entender que su fin único es recolectar y recuperar los datos a la mayor velocidad posible, por lo tanto la redundancia y duplicación de información no es un problema como con las demás bases de datos, por lo general para poderlas aprovechar al máximo permiten algún tipo de conectividad a bases de datos relacionales.

Un ejemplo habitual de transacción es el traspaso de una cantidad de dinero entre cuentas bancarias. Normalmente se realiza mediante dos operaciones distintas, una en la que se decrementa el saldo de la cuenta origen y otra en la que incrementamos el saldo de la cuenta destino. Para garantizar la atomicidad del sistema (es decir, para que no aparezca o desaparezca dinero), las dos operaciones deben ser atómicas, es decir, el sistema debe garantizar que, bajo cualquier circunstancia (incluso una caída del sistema), el resultado final es que, o bien se han realizado las dos operaciones, o bien no se ha realizado ninguna.

- **Bases de datos relacionales**

Éste es el modelo utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Tras ser postulados sus fundamentos en 1970 por Edgar Frank Codd, de los laboratorios IBM en San José (California), no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos. Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados "tuplas". Pese a que ésta es la teoría de las bases de datos relacionales creadas por Codd, la mayoría de las veces se conceptualiza de una manera más fácil de imaginar. Esto es pensando en cada relación como si fuese una tabla que está compuesta por *registros* (las filas de una tabla), que representarían las tuplas, y *campos* (las columnas de una tabla).

En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

El lenguaje más habitual para construir las consultas a bases de datos relacionales es SQL, *Structured Query Language* o *Lenguaje Estructurado de Consultas*, un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales.

Durante su diseño, una base de datos relacional pasa por un proceso al que se le conoce como normalización de una base de datos.

Durante los años 80 la aparición de dBASE produjo una revolución en los lenguajes de programación y sistemas de administración de datos. Aunque nunca debe olvidarse que dBase no utilizaba SQL como lenguaje base para su gestión.

- **Bases de datos multidimensionales**

Son bases de datos ideadas para desarrollar aplicaciones muy concretas, como creación de Cubos OLAP. Básicamente no se diferencian demasiado de las bases de datos relacionales (una tabla en una base de datos relacional podría serlo también en una base de datos multidimensional), la diferencia está más bien a nivel conceptual; en las bases de datos multidimensionales los campos o atributos de una tabla pueden ser de dos tipos, o bien representan dimensiones de la tabla, o bien representan métricas que se desean estudiar.

- **Bases de datos orientadas a objetos**

Este modelo, bastante reciente, y propio de los modelos informáticos orientados a objetos, trata de almacenar en la base de datos los *objetos* completos (estado y comportamiento).

Una base de datos orientada a objetos es una base de datos que incorpora todos los conceptos importantes del paradigma de objetos:

Encapsulación - Propiedad que permite ocultar la información al resto de los objetos, impidiendo así accesos incorrectos o conflictos.

Herencia - Propiedad a través de la cual los objetos heredan comportamiento dentro de una jerarquía de clases.

Polimorfismo - Propiedad de una operación mediante la cual puede ser aplicada a distintos tipos de objetos.

En bases de datos orientadas a objetos, los usuarios pueden definir operaciones sobre los datos como parte de la definición de la base de datos. Una operación (llamada función) se especifica en dos partes. La interfaz (o signatura) de una operación incluye el nombre de la operación y los tipos de datos de sus argumentos (o parámetros). La implementación (o método) de la operación se especifica separadamente y puede modificarse sin afectar la interfaz. Los programas de aplicación de los usuarios pueden operar sobre los datos invocando a dichas operaciones a través de sus nombres y argumentos, sea cual sea la forma en la que se han implementado. Esto podría denominarse independencia entre programas y operaciones.

SQL: 2003, es el estándar de SQL92 ampliado, soporta los conceptos orientados a objetos y mantiene la compatibilidad con SQL92.

- **Bases de datos documentales**

Permiten la indexación a texto completo, y en líneas generales realizar búsquedas más potentes. Taurus es un sistema de índices optimizado para este tipo de bases de datos.

- **Bases de datos deductivas**

Un sistema de base de datos deductiva, es un sistema de base de datos pero con la diferencia de que permite hacer deducciones a través de inferencias. Se basa principalmente en reglas y hechos que son almacenados en la base de datos. Las bases de datos deductivas son también llamadas bases de datos lógicas, a raíz de que se basa en lógica matemática. Este tipo de base de datos surge debido a las limitaciones de la Base de Datos Relacional de responder a consultas recursivas y de deducir relaciones indirectas de los datos almacenados en la base de datos.

1.1.7 .3.6 Lenguaje

Utiliza un subconjunto del lenguaje Prolog llamado Datalog el cual es declarativo y permite al ordenador hacer deducciones para contestar a consultas basándose en los hechos y reglas almacenados.

1.1.7 .3.7 Ventajas

- Uso de reglas lógicas para expresar las consultas.
- Permite responder consultas recursivas.
- Cuenta con negaciones estratificadas
- Capacidad de obtener nueva información a través de la ya almacenada en la base de datos mediante inferencia.
- Uso de algoritmos de optimización de consultas.
- Soporta objetos y conjuntos complejos.

1.1.7 .3.8 Desventajas

- Crear procedimientos eficaces de deducción para evitar caer en bucles infinitos.
- Encontrar criterios que decidan la utilización de una ley como regla de deducción.
- Replantear las convenciones habituales de la base de datos.

1.1.7 .3.9 Fases

- Fase de Interrogación: se encarga de buscar en la base de datos informaciones deducibles implícitas. Las reglas de esta fase se denominan reglas de derivación.
- Fase de Modificación: se encarga de añadir a la base de datos nuevas informaciones deducibles. Las reglas de esta fase se denominan reglas de generación.

1.1.7 .3.10 Interpretación

Encontramos dos teorías de interpretación de las bases de datos deductivas:

- Teoría de Demostración: consideramos las reglas y los hechos como axiomas.
- Los hechos son axiomas base que se consideran como verdaderos y no contienen variables. Las reglas son axiomas deductivos ya que se utilizan para deducir nuevos hechos.
- Teoría de Modelos: una interpretación es llamada modelo cuando para un conjunto específico de reglas, éstas se cumplen siempre para esa interpretación. Consiste en asignar a un predicado todas las combinaciones de valores y argumentos de un dominio de valores constantes dado. A continuación se debe verificar si ese predicado es verdadero o falso.

1.1.7 .3.11 Mecanismos

Existen dos mecanismos de inferencia:

- Ascendente: donde se parte de los hechos y se obtiene nuevos aplicando reglas de inferencia.
- Descendente: donde se parte del predicado (objetivo de la consulta realizada) e intenta encontrar similitudes entre las variables que nos lleven a hechos correctos almacenados en la base de datos.

Gestión de bases de datos distribuida (SGBD)

La base de datos y el software SGBD pueden estar distribuidos en múltiples sitios conectados por una red. Hay de dos tipos:

1. Distribuidos homogéneos: utilizan el mismo SGBD en múltiples sitios.
2. Distribuidos heterogéneos: Da lugar a los SGBD federados o sistemas multibase de datos en los que los SGBD participantes tienen cierto grado de autonomía local y tienen acceso a varias bases de datos autónomas preexistentes almacenados en los SGBD, muchos de estos emplean una arquitectura cliente-servidor.

1.1.7 .3.12 Conexiones a Bases de Datos

Las conexiones persistentes son enlaces SQL que no se cierran cuando la ejecución del script termina. El comportamiento de estas conexiones es el siguiente.

Cuando se invoca una conexión de este tipo, PHP comprueba si existe una conexión de este mismo tipo o por el contrario, se trata de una nueva conexión. En el caso de que exista, se procede a su uso, y en el caso de que no exista, la conexión se crea. Dos conexiones se consideran iguales cuando están realizadas sobre el mismo servidor, con el mismo usuario y la misma contraseña.

Pero en realidad, estas conexiones permanentes, no proporcionan ningún tipo de funcionalidad adicional frente a conexiones temporales, debido a la forma en que los servidores Web funcionan.

Aun así se utilizan debido a la eficiencia, debido al tiempo de establecimiento de la conexión, y debido a que si tienes una sola conexión sobre el servidor, irá mucho más rápido que si tienes 10 conexiones temporales, puesto que la carga que soporta es diferente.

1.1.7 .3.3 ¿Qué es MySQL?

MySQL es un sistema de administración de bases de datos.

A continuación las definiciones realizadas por algunos autores como:

Welling Luke; Thomson Laura (2005). “Una base de datos es una colección estructurada de datos. Esta puede ser desde una simple lista de compras a una galería de pinturas o el vasto monto de información en una red corporativa. Para agregar, accezar y procesar datos guardados en un computador, usted necesita un administrador como MySQL Server. Dado que los computadores son muy buenos manejando grandes cantidades de información, los administradores de bases de datos juegan un papel central en computación, como aplicaciones independientes o como parte de otras aplicaciones.”

- ✓ **MySQL es un sistema de administración relacional de bases de datos.**

Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido.

✓ **MySQL es software de fuente abierta.**

Fuente abierta significa que es posible para cualquier persona usarlo y modificarlo. Cualquier persona puede bajar el código fuente de MySQL y usarlo sin pagar. Cualquier interesado puede estudiar el código fuente y ajustarlo a sus necesidades. MySQL usa el GPL (GNU General Public License) para definir qué puede hacer y que no puede hacer con el software en diferentes situaciones. Si usted no se ajusta al GPL o requiere introducir código MySQL en aplicaciones comerciales, usted puede comprar una versión comercial licenciada.

1.1.7.3.3 Las características principales de MySQL.

✓ **Es un gestor de base de datos.**

Una base de datos es un conjunto de datos y un gestor de base de datos es una aplicación capaz de manejar este conjunto de datos de manera eficiente y cómoda.

✓ **Es una base de datos relacional.**

Una base de datos relacional es un conjunto de datos que están almacenados en tablas entre las cuales se establecen unas relaciones para manejar los datos de una forma eficiente y segura. Para usar y gestionar una base de datos relacional se usa el lenguaje estándar de programación SQL.

✓ **Es Open Source.**

El código fuente de MySQL se puede descargar y está accesible a cualquiera, por otra parte, usa la licencia GPL para aplicaciones no comerciales.

✓ **Es una base de datos muy rápida, segura y fácil de usar.**

Gracias a la colaboración de muchos usuarios, la base de datos se ha ido mejorando optimizándose en velocidad. Por eso es una de las bases de datos más usadas en Internet.

1.1.7.3.3.4 Conectándose y desconectándose al Servidor MySql.

Para conectarse al servidor, usualmente necesitamos de un nombre de usuario (login) y de una contraseña (password), y si el servidor al que nos deseamos conectar está en una máquina diferente de la nuestra, también necesitamos indicar el nombre o la dirección IP de dicho servidor. Una vez que conocemos estos tres valores, podemos conectarnos de la siguiente manera:

```
shell> mysql -h NombreDelServidor -u NombreDeUsuario -p
```

Cuando ejecutamos este comando, se nos pedirá que proporcionemos también la contraseña para el nombre de usuario que estamos usando.

Si la conexión al servidor MySQL se pudo establecer de manera satisfactoria, recibiremos el mensaje de bienvenida y estaremos en el prompt de MySql:

```
shell>MySQL -h casita -u root -p
```

```
Enter password: *****
```

```
Welcome to the MySQL monitor. Commands end with; or \g.
```

```
Your MySQL connection id is 5563 to server version: 3.23.41
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
```

```
MySQL> Este prompt nos indica que MySql está listo para recibir comandos.
```

Algunas instalaciones permiten que los usuarios se conecten de manera anónima al servidor corriendo en la máquina local. Si es el caso de nuestra máquina, debemos de ser capaces de conectarnos al servidor invocando a MySql sin ninguna opción.

Shell> Después de que nos hemos conectado de manera satisfactoria, podemos desconectarnos en cualquier momento al escribir "quit", "exit", o presionar CONTROL+D.

1.1.7.3.3.5 Servidor Web Apache.

Claudia Valdés, Miranda Cros; Enrique Rodríguez Álvarez (2004). “Apache es uno de los mejores servidores de Webs utilizados en la red internet desde hace mucho tiempo, únicamente le hace competencia un servidor de Microsoft, el IIS. Por lo que éste servidor es uno de los mayores triunfos del software libre, que tanto gusta a los usuarios de LINUX.”

Es un servidor de web flexible, rápido y eficiente, continuamente actualizado y adaptado a los nuevos protocolos (HTTP 1.1)

- Implementa los últimos protocolos, aunque se base en el HTTP / 1.1
- Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo y con la API de programación de módulos.
- Incentiva la realimentación de los usuarios, obteniendo nuevas ideas, informes de fallos y parches para solucionar los mismos.

La versión actual del apache es la 1.2.4 (1.3 ya está en beta). En la nueva versión se incluyen características como el soporte para Windows NT y Windows 95, así como la inclusión de cuatro dígitos en las fechas para evitar los problemas del año 2000.

1.1.7.3.3.6 El Protocolo HTTP.

Es el que da vida a internet, y gracias al cual, los clientes y servidores se permiten comunicar. Su funcionamiento básico consiste en que el cliente establece una conexión, utilizando el protocolo TCP, con el servidor; Y luego genera una petición, el servidor le responde y se cierra la conexión. En la versión (http 1.0), el cliente sólo podía invocar tres operaciones en el servidor:

1. GET => Para pedir una página.
2. HEAD => Para pedir la cabecera de un página.
3. POST => Para enviar datos a una URL.

1.1.7.3.3.7 Arquitectura del Servidor Apache.

El servidor Apache es un software que está estructurado en módulos. La configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías.

1. **Módulos Base:** Módulo con las funciones básicas del Apache.
2. **Módulos Multiproceso:** son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones.
3. **Módulos Adicionales:** Cualquier otro módulo que le añada una funcionalidad al servidor.

Las funcionalidades más elementales se encuentran en el módulo base, siendo necesario un módulo multiproceso para manejar las peticiones. Se han diseñado varios módulos multiproceso para cada uno de los sistemas operativos sobre los que se ejecuta el Apache, optimizando el rendimiento y rapidez del código.

El resto de funcionalidades del servidor se consiguen por medio de módulos adicionales que se pueden cargar. Para añadir un conjunto de utilidades al servidor, simplemente hay que añadirle un módulo, de forma que no es necesario volver a instalar el software.

1.1.7.3.3.8 Seguridad en la Web.

Dado el gran auge que hoy en día tiene Internet, su uso se ha masificado enormemente. Desde páginas meramente informativas hasta sitios interactivos usando tecnologías nuevas.

Empresas de diversa índole ya usan la Internet para comunicarse y el problema principal que surgió es la confiabilidad en que lo que se está comunicando no sea visto por personas que puedan hacer mal uso de dicha información.

Por ejemplo, las tiendas comerciales ya están dando la posibilidad de realizar compras por la Web, pero el principal talón de Aquiles lo constituye la inseguridad que causa dar un número de tarjeta de crédito para pagar la compra.

O cosas tan simples como cuando uno envía un mail y no querer que nadie lo lea sino el destinatario.

A raíz de todo esto surgieron tecnologías que persiguen mejorar la seguridad de todas estas comunicaciones.

1.1.7.3.3.9 Seguridad en la Transmisión.

La seguridad de este tipo se basa en el hecho de poder encriptar los mensajes que se envían por la red entre un servidor y un cliente y que solo ellos puedan descifrar los contenidos a partir de una clave común conocida solo por los dos.

Para llevar a cabo esta seguridad se crearon diversos protocolos basados en esta idea:

1. SSH: Usado exclusivamente en remplazo de telnet
2. SSL: Usado principalmente en comunicaciones de hipertexto pero con posibilidad de uso en otros protocolos
3. TSL: Es del mismo estilo del anterior.
4. HTTPS: Usado exclusivamente para comunicaciones de hipertexto

a. SSH.

Este protocolo fue diseñado para dar seguridad al acceso a computadores en forma remota.

Cumple la misma función que telnet o rlogin pero además, usando criptografía, logra seguridad con los datos.

A diferencia de telnet u otro servicio similar, SSH utiliza el puerto 22 para la comunicación y la forma de efectuar su trabajo es muy similar al efectuado por SSL.

Para su uso se requiere que por parte del servidor exista un demonio que mantenga continuamente en el puerto 22 el servicio de comunicación segura, el sshd.

El cliente debe ser un software tipo TeraTerm o Putty que permita al hacer pedidos a este puerto 22 de forma cifrada.

La forma en que se entabla una comunicación es en base la misma para todos los protocolos seguros.

El cliente envía una señal al servidor pidiéndole comunicación por el puerto 22.

El servidor acepta la comunicación en el caso de poder mantenerla bajo encriptación mediante un algoritmo definido y le envía la llave pública al cliente para que pueda descifrar los mensajes.

El cliente recibe la llave teniendo la posibilidad de guardar la llave para futuras comunicaciones o destruirla después de la sesión actual.

Se recomienda que si se está en un computador propio, la clave sea guardada, en otro caso, destruirla.

b. SSL (Secure Socket Layer) y TLS (Transport Layer Secure).

El protocolo SSL fue desarrollado por Netscape para permitir confidencialidad y autenticación en Internet. SSL es una capa por debajo de HTTP y tal como lo indica su nombre esta a nivel de socket por lo que permite ser usado no tan solo para proteger documentos de hipertexto sino también servicios como FTP, SMTP, TELNET entre otros.

La idea que persigue SSL es encriptar la comunicación entre servidor y cliente mediante el uso de llaves y algoritmos de encriptación.

El protocolo TLS está basado en SSL y son similares en el modo de operar.

Es importante señalar que ambos protocolos se ejecutan sobre una capa de transporte definida, pero no determinada. Esto indica que pueden ser utilizados para cualquier tipo de comunicaciones. La capa de transporte más usada es TCP sobre la cual pueden implementar seguridad en HTTP.

Como punto de diferencia se puede mencionar que existen protocolos implementados sobre la capa de red, por ejemplo sobre IP. Tal es el caso de IPSec.

¿De qué están compuestos?

Estos protocolos se componen de dos capas: el Record Protocol y el Handshake Protocol.

El Record Protocol es la capa inmediatamente superior a TCP y proporciona una comunicación segura. Principalmente esta capa toma los mensajes y los codifica con algoritmos de encriptación de llave simétrica como DES, RC4 aplicándole una MAC (Message Authentication Code) para verificar la integridad, logrando así encapsular la seguridad para niveles superiores.

El Handshake protocol es la capa superior a la anterior y es usada para gestionar la conexión inicial.

c. ¿Cómo funcionan?

En resumidas cuentas, después que se solicita una comunicación segura, servidor y el cliente se deben poner de acuerdo en cómo se comunicaran (SSL Handshake) para luego comenzar la comunicación encriptada. Luego de terminada la transacción, SSL termina.

Solicitud de SSL: Típicamente este proceso ocurre en el momento que un cliente accede a un servidor seguro, identificado con "https://...". Pero como se mencionó, no

necesariamente es usado para HTTP. La comunicación se establecerá por un puerto distinto al utilizado por el servicio normalmente. Luego de esta petición, se procede al SSL Handshake.

SSL Handshake: En este momento, servidor y cliente se ponen de acuerdo en varios parámetros de la comunicación. Se puede dividir el proceso en distintos pasos:

Client Hello: El cliente se presenta. Le pide al servidor que se presente (certifique quien es) y le comunica que algoritmos de encriptación soporta y le envía un número aleatorio para el caso que el servidor no pueda certificar su validez y que aun así se pueda realizar la comunicación segura.

Server Hello: El servidor se presenta. Le responde al cliente con su identificador digital encriptado, su llave pública, el algoritmo que se usará, y otro número aleatorio. El algoritmo usado será el más poderoso que soporte tanto el servidor como el cliente.

Aceptación del cliente: El cliente recibe el identificador digital del servidor, lo descripta usando la llave pública también recibida y verifica que dicha identificación proviene de una empresa certificadora segura. Luego se procede a realizar verificaciones del certificado (identificador) por medio de fechas, URL del servidor, etc. Finalmente el cliente genera una llave aleatoria usando la llave pública del servidor y el algoritmo seleccionado y se la envía al servidor.

Verificación: Ahora tanto el cliente y el servidor conocen la llave aleatoria (El cliente la generó y el servidor la recibió y descriptó con su llave privada). Para asegurar que nada ha cambiado, ambas partes se envían las llaves. Si coinciden, el Handshake concluye y comienza la transacción.

d. HTTPS es la versión segura del protocolo HTTP.

El sistema HTTPS utiliza un cifrado basado en las Secure Socket Layers (SSL) para crear un canal cifrado (cuyo nivel de cifrado depende del servidor remoto y del navegador utilizado por el cliente) más apropiado para el tráfico de información sensible que el protocolo HTTP. Cabe mencionar que el uso del protocolo HTTPS no

impide que se pueda utilizar HTTP. Es aquí, cuando nuestro navegador nos advertirá sobre la carga de elementos no seguros (HTTP), estando conectados a un entorno seguro (HTTPS).

Los protocolos https son utilizados por navegadores como: Safari (navegador), Internet Explorer, Mozilla Firefox, Opera, entre otros.

Es utilizado principalmente por entidades bancarias, tiendas en línea, y cualquier tipo de servicio que requiera el envío de datos personales o contraseñas.

El puerto estándar para este protocolo es el 443.

Para conocer si una página web que estamos visitando, utiliza el protocolo https y es, por tanto, segura en cuanto a la transmisión de los datos que estamos transcribiendo, debemos observar si en la barra de direcciones de nuestro navegador, aparece https al comienzo, en lugar de http.

Algunos navegadores utilizan un icono en la barra de estado (parte inferior de la ventana), indicando la existencia de un protocolo de comunicaciones seguro.

1.1.7 .3.4 Instalación: SQLYog 8.12

Para instalar el SQLYog, hay que descargárselo en:

<http://www.webyog.com/en/downloads.php#sqlyog> Y escoger de la CommunityEdition, el Binaries – Windows más actual.



Grafico Nº 4: Instalación SQLYog

Al clicar el SQLyog, hay que pulsar a 'Next', después escoger el 'I accept the terms in the License Agreement', pulsar 'Next', pulsar 'Next', y para finalizar pulsar 'Install', 'Next', y 'Finish'.



Grafico N°5: Instalación SQLYog

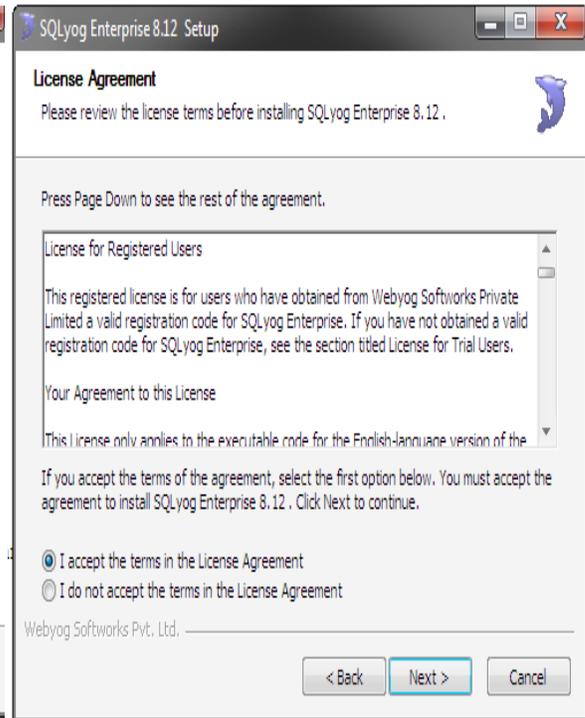


grafico N°6: Instalación SQLYog

Al iniciarlo, hay que especificar donde está la base de datos. Pulsamos en el botón 'Install'.

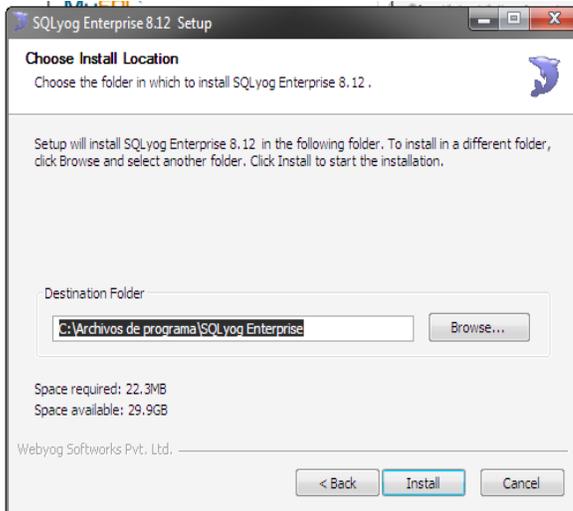


Grafico Nº 7: Instalación SQLYog

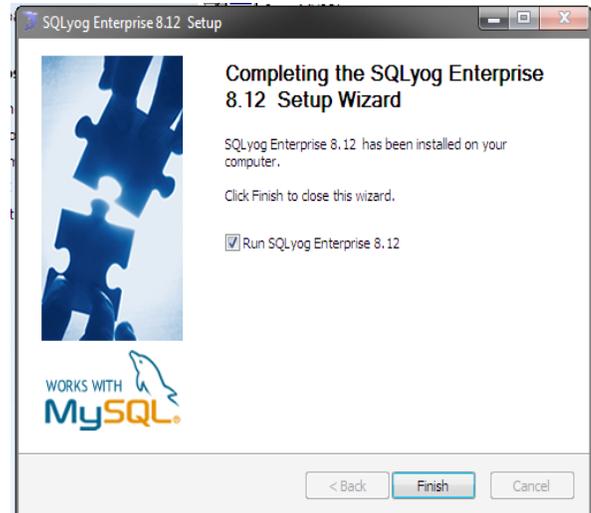


Grafico Nº 8: Instalación SQLYog

SQLyog es un administrador de bases de datos MySQL para Windows que, recientemente, ha liberado su versión para código abierto, gratuita para uso no comercial.

WAMP es el acrónimo usado para describir un sistema de infraestructura de internet que usa las siguientes herramientas:

- Windows, como sistema operativo;
- Apache, como servidor web;
- MySQL, como gestor de bases de datos;

El uso de un WAMP permite servir páginas html a internet, además de poder gestionar datos en ellas, al mismo tiempo un WAMP, proporciona lenguajes de programación para desarrollar aplicaciones web.

CAPITULO II

Marco Teórico

2.1 Antecedentes y Estudios Previos

Al presente trabajo de investigación, no le antecede otro proyecto similar después de haber realizado la investigación en la biblioteca virtual de la FAFI donde este trabajo surge en base al análisis realizado de la problemática dentro del centro de salud.

Durante el tiempo de revisión de datos se observó que existen grandes desfases e incongruencias en la gestión de ingreso de registro de los pacientes que asiste a este centro de salud, la cual no contiene un software que permita llevar y archivar la información rápida y eficaz.

Se ha observado previamente que en el centro de salud, no contiene un sistema de gestión de análisis clínico que permita automatizar todos los registros de los pacientes de acuerdo al análisis de los exámenes, mejorando el tiempo de repuesta de acuerdo a las necesidades del laboratorista y de los pacientes.

Bajo estos antecedentes, el proyecto que se quiere implementar es de un sistema de análisis clínico donde el cual se pueda guardar, modificar eliminar registros, consultas generales, consultas específicas, y reportes; de este modo se podrá mejora la atención al paciente.

2.2 Fundamentación Teórica

2.2.1 Introducción

En el campo de la administración muchos gerentes han fracasado por el desconocimiento del avance administrativo, actualmente está nos ofrece un nuevo mundo de cómo administrar nuestra empresa, poniéndolo en práctica todas las herramientas que se requieren para brindar servicios de calidad, si bien es cierto este trabajo investigativo' lo ejecutamos en el centro de salud Echeandía en donde hemos encontrado una problemática debido al desconocimiento de una buena administración, por esta razón le invitamos al amigo lector e interesados en adquirir

información confiable, prestar atención en el contenido que se encuentra detallado según los capítulos que requiera el investigador porque en pleno siglo XXI tenemos instancias públicas con gerentes que desconocen la verdadera administración.

Además como investigadores es nuestro deber y obligación orientar a todo el personal involucrado al manejo de la guía práctica en administración hospitalaria del Cantón Echeandía, para lograr una buena administración y dar servicios de calidad.

2.3 Definición De Términos

El “**Laboratorio clínico**” es el lugar donde los profesionales de laboratorio de diagnóstico clínico (Tecnólogo Médico, Bioquímicos, Químicos Farmacéuticos, Bioanálistas, Químicos Bacteriólogos Parasitólogos y Médicos) realizan análisis clínicos que contribuyen al estudio, prevención, diagnóstico y tratamiento de los problemas de salud de los pacientes. También se le conoce como Laboratorio de Patología clínica. Los laboratorios de análisis clínicos, de acuerdo con sus funciones, se pueden dividir en:

1. **Laboratorios de Rutina o de seguimiento.** Los laboratorios de rutina tienen cuatro departamento básicos: Hematología, Inmunología, Microbiología y Química Clínica (o Bioquímica).

Los laboratorios de rutina pueden encontrarse dentro de un hospital o ser externos a éste. Los laboratorios hospitalarios, con frecuencia tienen secciones consideradas de urgencia, donde se realizan estudios que servirán para tomar decisiones críticas en la atención de los pacientes graves. Estudios tales como citometría hepática, tiempos de coagulación, glucemia, urea, creatinina y gases sanguíneos.

2. **Laboratorios de Especialidad.** En los laboratorios de pruebas especiales se realizan estudios más sofisticados, utilizando metodologías como amplificación de ácidos nucleicos, estudios cromosómicos, citometría de flujo y cromatografía de alta resolución, entre otros. Estas pruebas requieren

instalaciones y adiestramiento especial del personal que las realiza. Con frecuencia, estos laboratorios forman parte de programas de investigación.

2.3.1 Estructuración del centro de salud

2.3.1.1 Ubicación Geográfica

La ubicación de la jefatura del Área 4 Echeandía se encuentra en la hoya de Chimbo, ubicada en la Avenida 5 de Octubre SN, en una colina alta respecto al centro urbano, las calles de acceso presentan una declinación muy pronunciada y solo una es adoquinada, el resto presentan dificultad para su asenso tanto vehicular como peatonal. Cercano a la unidad de salud está el estadio cantonal, colinda con la escuela 5 de Octubre, está alejada de la central de cuerpo de bomberos la misma que se ubica en la zona urbana. Como institución el Centro está ubicado a 65 Km. Del Hospital Alfredo

Noboa Montenegro de Guaranda, a 18 kilómetros del hospital de Ventanas y a 120 Km. de los hospitales de Guayaquil.

Su ámbito territorial comprende 104 Comunidades distribuidas en 8 unidades Operativas que son:

- Centro de Salud Echeandía (Jefatura de Área)
- Subcentro de Salud Sabanetillas
- Subcentro de Salud Camarón
- Subcentro de Salud Chazo Juan
- Subcentro de Salud La Palma
- Subcentro de Salud Las Naves
- Subcentro de Salud Las Mercedes
- Subcentro de Salud San Luís de Pambil

En el contexto cantonal el Cantón Echeandía se encuentra formando parte de la hoya de Chimbo, ubicado en el sub trópico de la Provincia Bolívar, presenta una

naturaleza diferente a los demás rincones de nuestra Patria. El Clima es el cálido que va desde 18 a 30 grados y se caracteriza por un nivel de humedad que llega a un 80%. Las estaciones son bien marcadas la lluviosa es la más larga que va de Diciembre a Junio y la seca o verano que va de Junio a Diciembre. Está atravesado por el caudaloso río Soloma y en época invernal produce inundaciones leves y frecuentes las que al agravarse, pueden producir desastres.

2.3.1.2 Relación de los Antecedentes y Trascendencia de la Entidad

1.- La administración como área de salud se inició a partir de Noviembre de 1994; El actual centro de Salud Echeandía está prestando sus servicios desde 20 de octubre del 2002.

El personal paulatinamente se va incrementando y se organizando en los diferentes departamentos por servicios:

- Dirección
- Coordinación
- Gestión Financiera (Presupuesto y Contabilidad, movimiento de caja)
- Desarrollo Organizacional (Recursos humanos, Servicios Institucionales y Farmacia)
- Gestión de Enfermería.
- Valor agregado (Estadística, Salud comunitaria, Consulta externa, Emergencia , Epidemiología, promoción de los servicios de salud , Oferta y demanda , Control de normas de Salud , Laboratorio, Ecografía, Vigilancia sanitaria)

2.3.2 Estructura Organización por Proceso

1.- Copias de los Organigramas



Tabla: Nº 3: Estructura organizacional por proceso

2.3.3 Obtener Manuales de Procedimientos Respecto a las Actividades.

Actividades principales

- Consulta Externa
- Obstetricia
- Odontología
- Laboratorio
- Emergencia.

Actividades secundarias

- Vacuna
- Post consulta.

En este caso no se tiene conocimiento de manuales para el procedimiento de cada una de las actividades, el mismo que no facilita un mejor servicio.

Se recomienda que las autoridades de la institución tomen consciencia de la falta de cada uno de los procesos adecuados y lo implemente en un tiempo no muy tardío, para que pueda prestar de mejor manera los servicios.

2.3.4 Funcionamiento del laboratorio

a.- Obtener Descripciones de Funciones en el Caso de no Disponer de Manuales.

Dr. Xavier Garzón	Director de la Institución
Ing. Diana Chicaiza	Financiera
Ing. Valeria Veloz	Pagadora.
Lc. Jaime Villacrés	Recursos humanos.
Sr. Danny Monar	Administrativo
Sra. Shelia Escudero	Secretaria contratada.
Srta. LenisVerdezoto	Estadística

En este departamento el personal es por contratos temporales.

La recomendación es que realicen gestiones antes sus autoridades superiores para crear partidas o las contrataciones tengan mayor tiempo sin descuidar el rendimiento de cada uno del personal, para que los procesos y actividades puedan tener continuidad y no sufran cambios durante la transición.

b. Distribución de los Empleados Dentro de la Entidad.

CONTRATO COLECTIVO

Sra. Isabel Camacho Narváez	Auxiliar de odontología
Sr. Rene Carvajal Vásconez	Auxiliar Administrativo de salud
Sra. Sara Melendez Cisneros	Auxiliar de enfermería
Sra. Olga PuninaLeyme	Auxiliar de odontología

Sra. Marola Ramírez Merelo	Auxiliar de odontología
Sr. Franklin Escudero	Inspector Sanitario
Sr. Evelina Guerrero Quiño	Auxiliar de odontología
Sra. Margarita Borja Capuz	Auxiliar de odontología
Sra. Carmelina Rodríguez	Auxiliar de enfermería
Sr. Cristian Llanos Ortiz	Auxiliar de enfermería

No se tuvo la información orgánica de cada uno del personal de acuerdo a las especialidades tanto médicas como administrativas. p

c.- Personal Administrativo

Castillo Álvarez	Profesional 5
GonzalezMinaya Walter	Profesional
Pereira Valero Hortensia	Asistente Administrativo
Riera Verdezoto Mary	Profesional 3
Soliz Tapia Gladis	Profesional 4
ChacánChacán Víctor	Profesional 4
Changoluisa Laura	Profesional 4
García Alarcón Franklin	Profesional 4
Roldán Lara Raúl	Profesional 5
Saltos Maldonado Eda	Profesional 3
Caminos Verdezoto Gladis	Auxiliar de Servicios

2.3.5 Estudio de Misión y Visión del Centro de Salud.

Misión

Desarrollar técnicas y gerencialmente, los servicios de la salud de zona de influencia para brindar atención de calidad, eficiencia y equidad en el contexto de la modernización del estado con estrategias de participación y control social.

Visión

El área de salud de Echeandía garantiza a la población mejores condiciones de salud a través de una prestación de servicios con calidad, equidad, eficiencia y efectividad manejando criterios de prioridad a grupos más vulnerables contando con capital humano capacitado, motivado y utilizado, la tecnología acorde al riesgo epidemiológico de la población, lo que se reflejará en la disminución de los índices de infección de la morbilidad y mortalidad.

2.3.5.1 Objetivos Estratégicos.

Para poder cumplir con la visión y misión propuestas por el área tenemos los siguientes objetivos estratégicos.

- Capacitar el personal.
- Mejorar la calidad de atención.

2.3.6 Estructura Orgánica

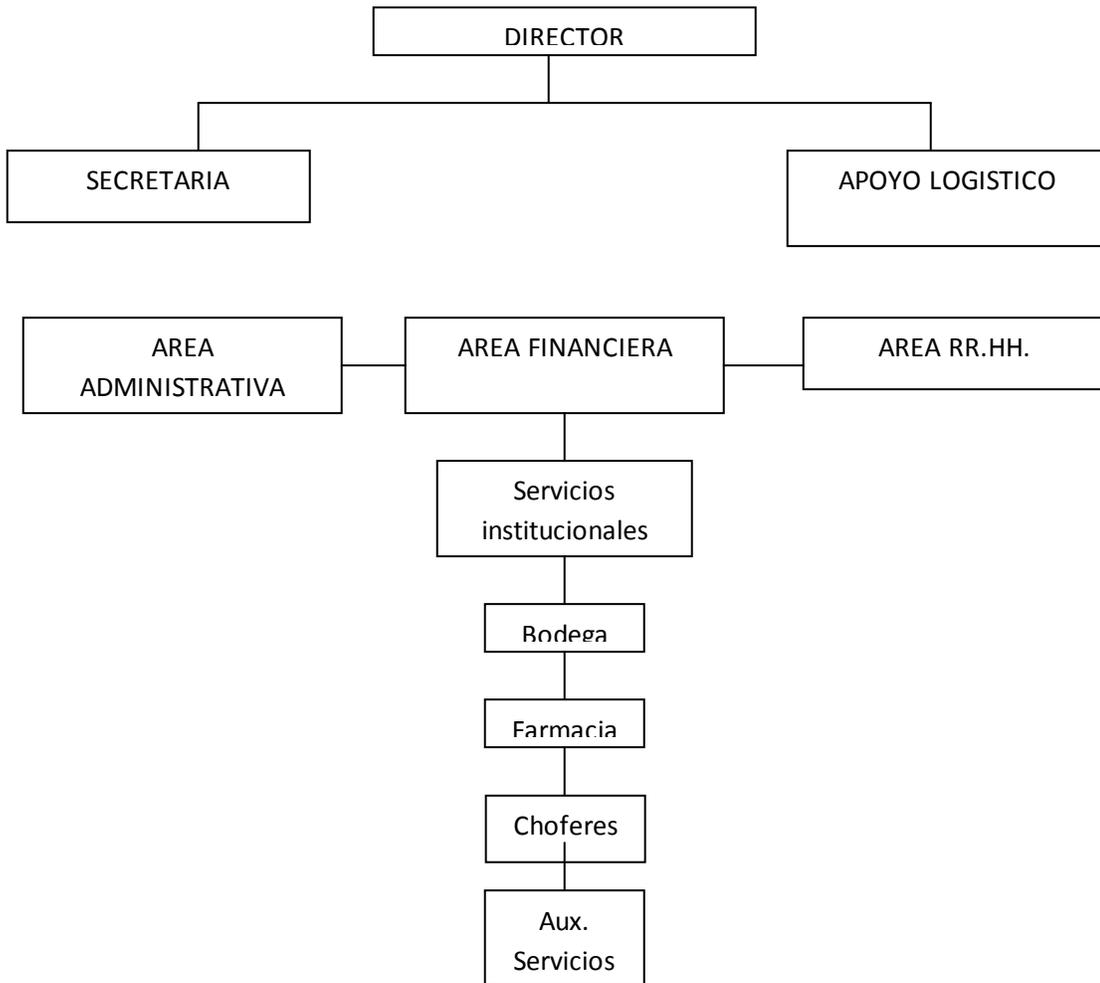


Tabla: Nº 4: Estructura Orgánica

2.3.7 Entrada del centro de salud

Fotografía Del Centro De Salud



Foto N°1 del Centro de Salud de Echeandía

Llegada Principal



Foto N° 2 Llegad Principal de Centro de Salud de Echeandía

Pasillo al Laboratorio



Foto N°3 Pasillo al Laboratorio

Entrada a la Sala del Laboratorio



Foto N°4 Sala del Laboratorio

Oficina



Foto N°5 Oficina del Laboratorio

Análisis Clínico y Laboratorista



Foto N° 6 Oficina del Laboratorio



Foto N° 7 Oficina del Laboratorio

CAPITULO III

Marco Hipótesis

3.1 Planteamientos De Hipótesis Y Variables

3.1.1 Planteamiento de Hipótesis

La falta de un sistema de gestión de análisis clínico, causa perdida de registros y tiempo.

3.1.2 Planteamiento de Variables

- La pérdida de registros y tiempo se debe a la falta de un sistema informático.
- El proceso de registro del laboratorio clínico es deficiente.
- La entrega de registro a los pacientes proviene incomodidad.

3.2 Operacionalización de Variables e Indicadores

3.2.1. Hipótesis

Con el desarrollo de un sistema de gestión facilitaremos el ingreso de datos del laboratorio de examen clínico. Un sistema operativo es un programa destinado a permitir la comunicación del usuario con un computador y gestionar sus recursos de una forma eficaz.

3.2.2. Variables

Variable Independiente:

Ingreso de datos del laboratorio de examen clínico.

Variable Dependiente:

Sistema de gestión

CAPITULO IV

Marco Metodológico

4.1 Diseño de la Investigación

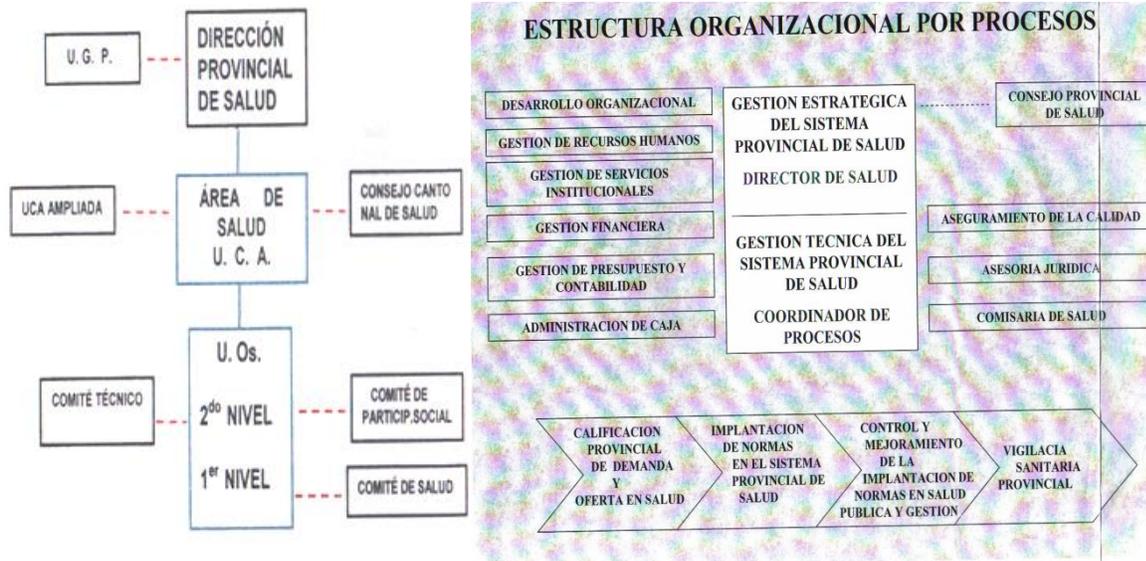


Tabla: Nº 5 Pantalla del diseño de la investigación

4.1.1 Modalidad de la Investigación

La modalidad con la que se va a realizar la investigación del proyecto será básicamente de estas dos metodologías:

- **Cualitativa:** La **investigación cualitativa** o **metodología cualitativa** es un método de investigación usado principalmente en las ciencias sociales que se basa en cortes metodológicos basados en principios teóricos tales como la fenomenología, hermenéutica, la interacción social empleando métodos de recolección de datos que son no cuantitativos, con el propósito de explorar las relaciones sociales y describir la realidad tal como la experimentan los

correspondientes. La investigación cualitativa requiere un profundo entendimiento del comportamiento humano y las razones que lo gobiernan.

Se elige esta modalidad porque de esta forma determinaremos las características que contenga el problema planteado.

- **Cuantitativa:** La metodología cuantitativa es aquella que permite examinar los datos de manera científica, o de manera más específicamente en forma numérica, generalmente con ayuda de herramientas del campo de la estadística.

Con esta modalidad se podrá confirmar las características que contenga el problema ya planteado en relación a datos numéricos recolectados.

4.1.2 Tipo de la Investigación

La investigación exploratoria ; No intenta dar explicación respecto del problema, sino sólo recoger e identificar antecedentes generales, números y cuantificaciones, temas y tópicos respecto del problema investigado, sugerencias de aspectos relacionados que deberían examinarse en profundidad en futuras investigaciones. Su objetivo es documentar ciertas experiencias, examinar temas o problemas poco estudiados o que no han sido abordadas antes. Por lo general investigan tendencias, identifican relaciones potenciales entre variables y establecen el “tono” de investigaciones posteriores más rigurosas.

Se efectúan, normalmente, cuando el objetivo es examinar un tema o problema de investigación poco estudiado o que no ha sido abordado antes.

Esta investigación se llevara a cabo debido a que se explorará el proceso de cómo se recibe y guarda la información acerca de todos los registros de los pacientes en el Centro de Salud de Echeandía.

4.1.3 Población y Muestra

Población estadística, en estadística, también llamada universo o colectivo, es el conjunto de elementos de referencia sobre el que se realizan las observaciones.

El número de elementos o sujetos que componen una población estadística es igual o mayor que el número de elementos que se obtienen de ella en una muestra (**n**).

En estadística una muestra estadística (también llamada muestra aleatoria o simplemente muestra) es un subconjunto de casos o individuos de una población estadística.

Población

La población en el cual enfocaremos la investigación será a la población de Echeandía debido a que ellos son los que asisten a este Centro de Salud.

Promedio de pacientes que va a visitar el laboratorio es de un total de 36 diarias; dando un total de 180 a la semana; entre este grupo **100** son personas de entre mayor de 16 años y menores de 70 años.

Muestra

En estadística una muestra estadística (también llamada muestra aleatoria o simplemente muestra) es un subconjunto de casos o individuos de una población estadística.

4.1.4 Métodos, Técnicas e Instrumentos

Método

El método que se usara es Inductivo-Deductivo ya que por este medio se podrá deducir la posible solución generalizada en base a una solución en particular;

resaltando que se partirá y se analizará en base a hechos conocidos para dar solución al problema.

Técnica

Las técnicas que se emplearan en la investigación serán:

- Encuesta
- Entrevista

Instrumentos

Los instrumentos serán los siguientes:

- Cuestionario
- Guía de entrevista
- Cuestionario

Encuesta:

Encuesta realizada a los pacientes del laboratorio clínico del Centro de Salud de la ciudad de Echeandía

1.- ¿Considera usted que hay bastante afluencia en el laboratorio clínico?

CUADRO N° 1		
	ENCUESTA	%MUESTRA
SI	75	93,75%
NO	5	6,25%
TOTAL	80	100%

Tabla N° 6: Encuesta primera pregunta

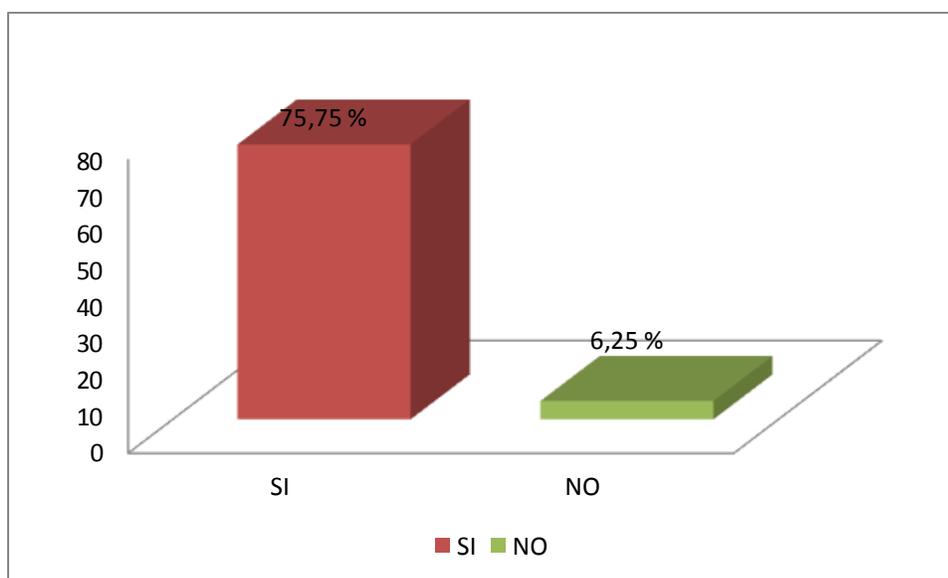


Gráfico N° 9 ENCUESTA; PRIMERA PREGUNTA

Los encuestados piensan que hay bastante afluencia en el laboratorio.

2.- ¿Cree usted que hay un desorden al momento de ingresar los registros de los pacientes?

CUADRO N° 2		
	ENCUESTA	% MUESTRA
SI	70	87,50%
NO	10	12,50%
TOTAL	80	100%

Tabla N° 7: Encuesta segunda pregunta

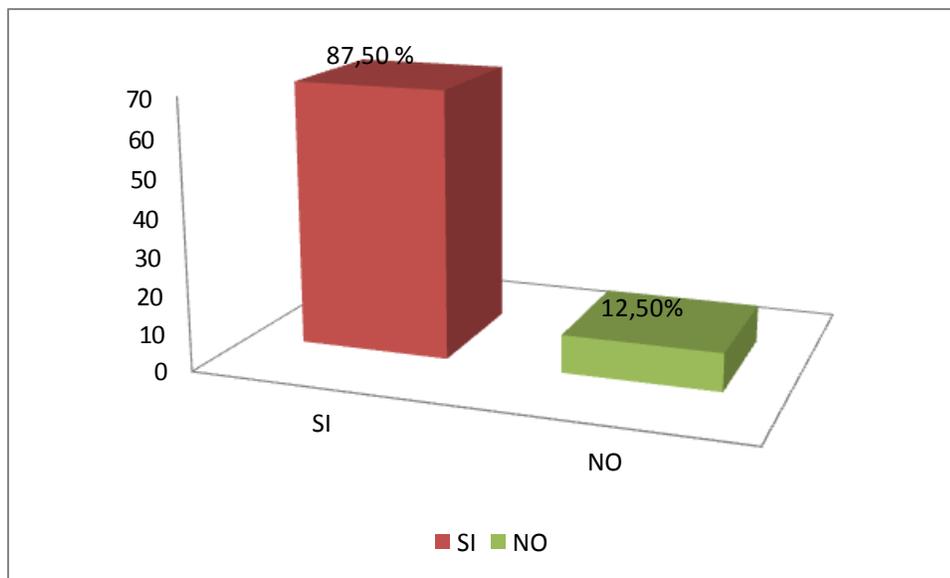


Grafico N° 10 ENCUESTA; SEGUNDA PREGUNTA

La mayoría de los encuestados piensan que si hay desorden en el registro de pacientes.

3.- ¿Cree usted que hay una demora al momento de retirar los exámenes del laboratorio?

CUADRO Nº 3		
	ENCUETA	% MUESTRA
SI	60	75%
NO	20	25%
TOTAL	80	100%

Tabla Nº 8: Encuesta tercera pregunta

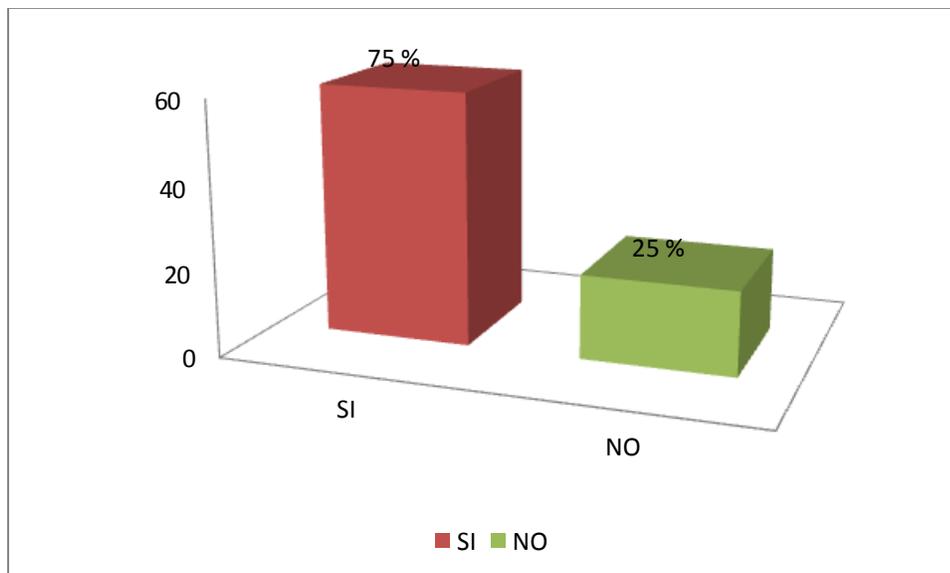


Grafico Nº 11 ENCUESTA; TERCERA PREGUNTA

Los encuestados piensas que hay demora al momento de retirar los exámenes.

4.- ¿Considera usted que debido al manejo manual de la documentación las actividades del laboratorio se torna muy lentas?

CUADRO N° 4		
	ENCUESTA	% MUESTRA
SI	65	81,25%
NO	15	18,75%
TOTAL	80	100%

Tabla N° 9: Encuesta cuarta pregunta

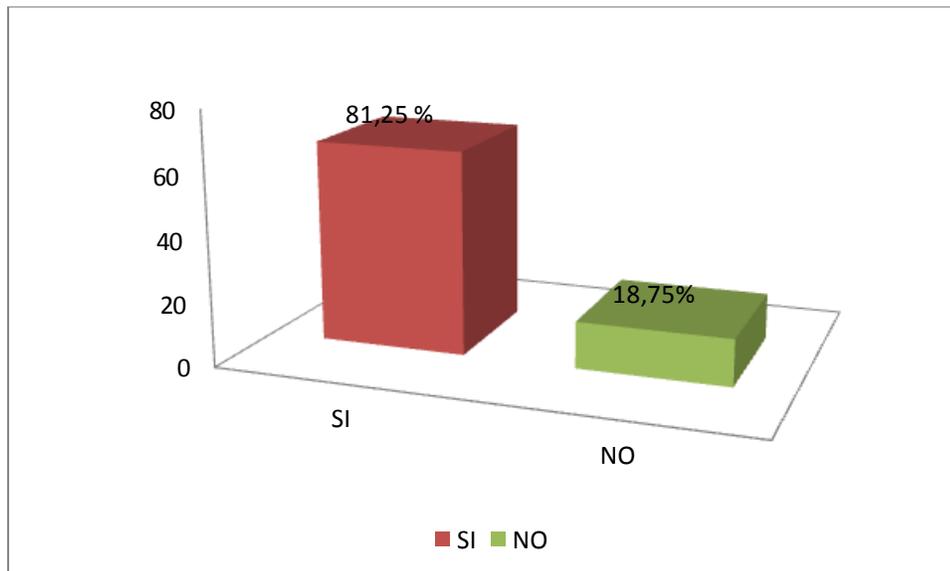


Grafico N° 12 ENCUESTA; CUARTA PREGUNTA

La mayoría de los encuestados piensan que debido a que se lleva un registro manual las actividades se tornan lentas.

5.- ¿Considera usted que se maneja gran cantidad de registro del laboratorio?

CUADRO N° 5		
	ENCUESTA	% MUESTRA
SI	67	83,75%
NO	13	16,25%
TOTAL	80	100%

Tabla N° 10: Encuesta quinta pregunta

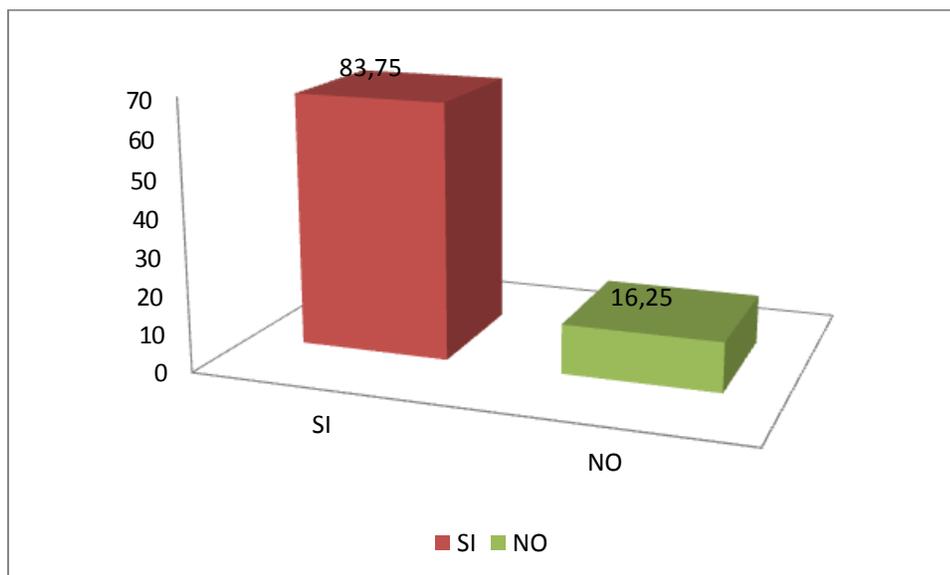


Gráfico N° 13 ENCUESTA; QUINTA PREGUNTA

Los encuetados si piensan que se maneja gran cantidad de registros en el laboratorio.

6.- ¿Cree usted que con un sistema adecuado se ordenaría de una mejor forma los registros del laboratorio?

CUADRO N° 6		
	ENCUESTA	% MUESTRA
SI	68	85%
NO	12	15%
TOTAL	80	100%

Tabla N°11: Encuesta sexta pregunta

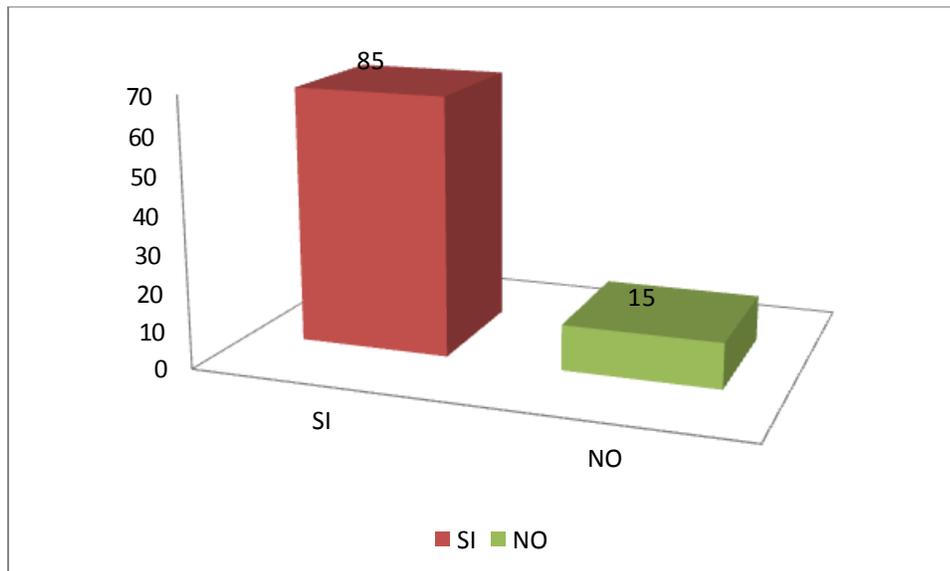


Grafico N° 14 ENCUESTA; SEXTA PREGUNTA

La mayoría de los encuestados piensan que sería conveniente un sistema para ordenar los registros en el laboratorio.

7.- ¿Cree usted que un sistema de informático ayudaría a mejorar la atención al paciente?

CUADRO N ^o 7		
	ENCUESTA	%MUESTRA
SI	68	85%
NO	12	15%
TOTAL	80	100%

Tabla N^o 12: Encuesta séptima pregunta

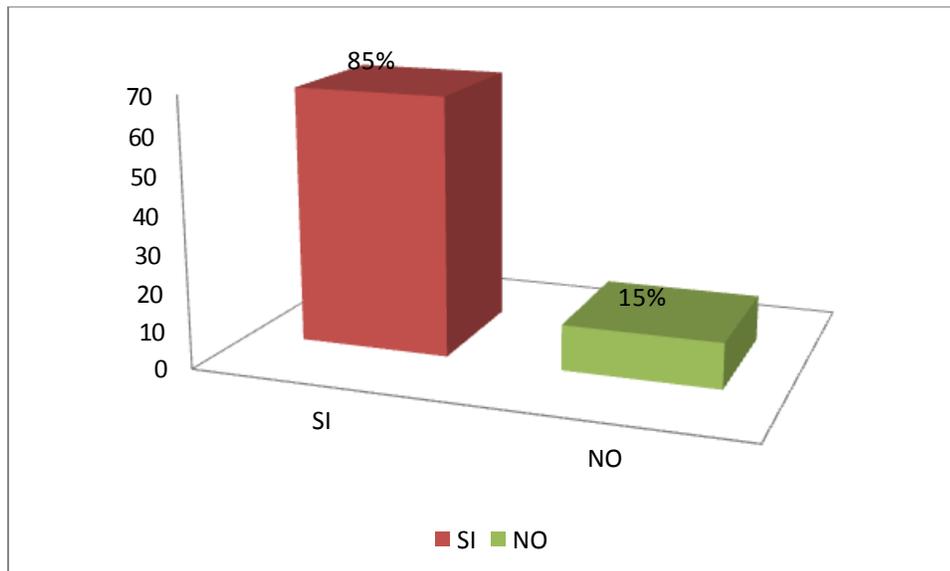


Grafico N^o 15 ENCUESTA; SEPTIMA PREGUNTA

Los encuestados están de acuerdo que con un portal web se mejoraría la atención.

8.- ¿Cree usted que mediante un sistema web mejoraría el tiempo de respuesta?

CUADRO N° 8		
	ENCUESTA	%MUESTRA
SI	70	87,5%
NO	10	12,5%
TOTAL	80	100%

Tabla N° 13: Encuesta séptima pregunta

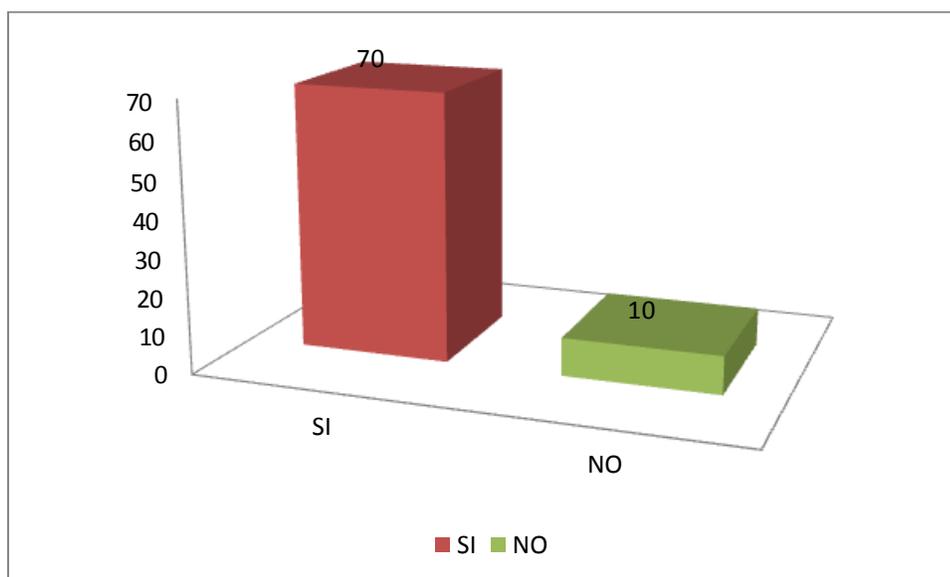


Gráfico N° 16 ENCUESTA; OCTAVA PREGUNTA

La mayoría de los encuestados está de acuerdo con un sistema web para mejorar los trámites en el laboratorio.

9.- ¿Está usted de acuerdo que se registre los análisis en un sistema informático para mejorar la agilidad del laboratorista?

CUADRO N° 9		
	ENCUESTA	%MUESTRA
SI	70	87,5
NO	10	12,5
TOTAL	80	100%

Tabla N° 14: Encuesta novena pregunta

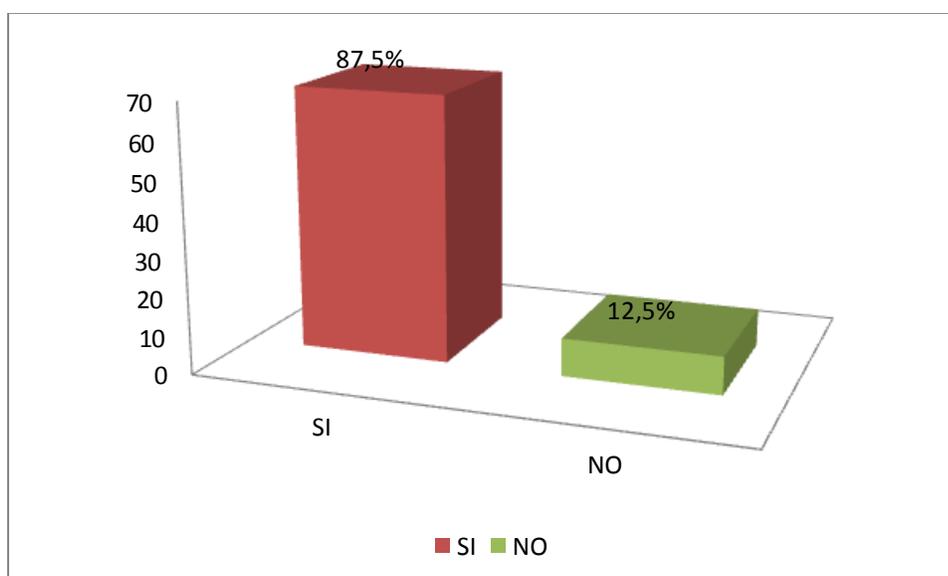


Grafico N° 17 ENCUESTA; NOVENA PREGUNTA

Los encuestados están de acuerdo con un portal web para agilizar el trabajo del laboratorista.

10.- ¿Sería necesario la capacitación informática en caso de implementar un sistemas informático?

CUADRO N ^o 10		
	ENCUESTA	%MUESTRA
SI	75	93,75
NO	5	6,25
TOTAL	80	100%

Tabla N^o 15: Encuesta decima pregunta

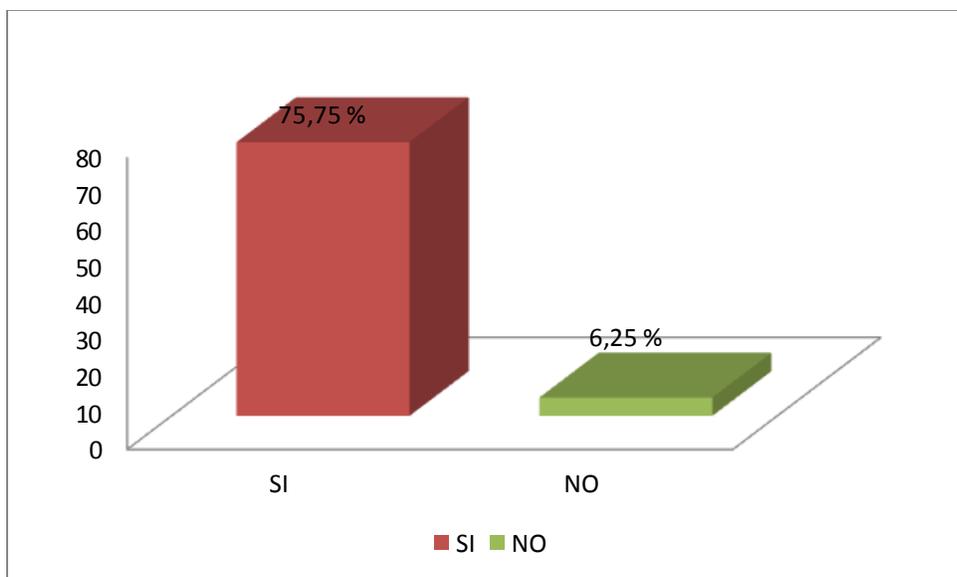


Grafico N^o 18 ENCUESTA; DECIMA PREGUNTA

La mayoría de los encuestados piensas que sería muy buena una capacitación informática en caso de la implementación de un sistema.

4.2 Procesamiento y Análisis

4.2.1 Diagramas de Caso de uso

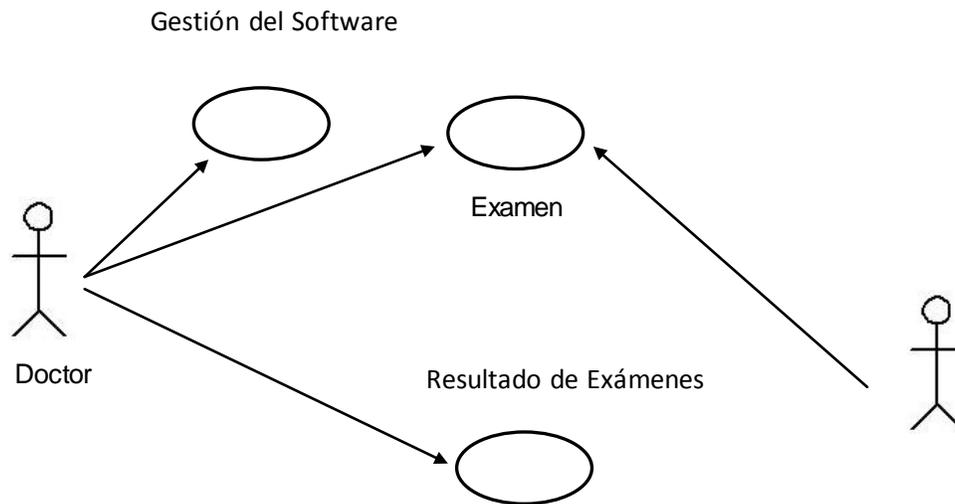


Gráfico N° 19: Diagramas de caso de uso

4.2.2 Diagrama de Secuencia

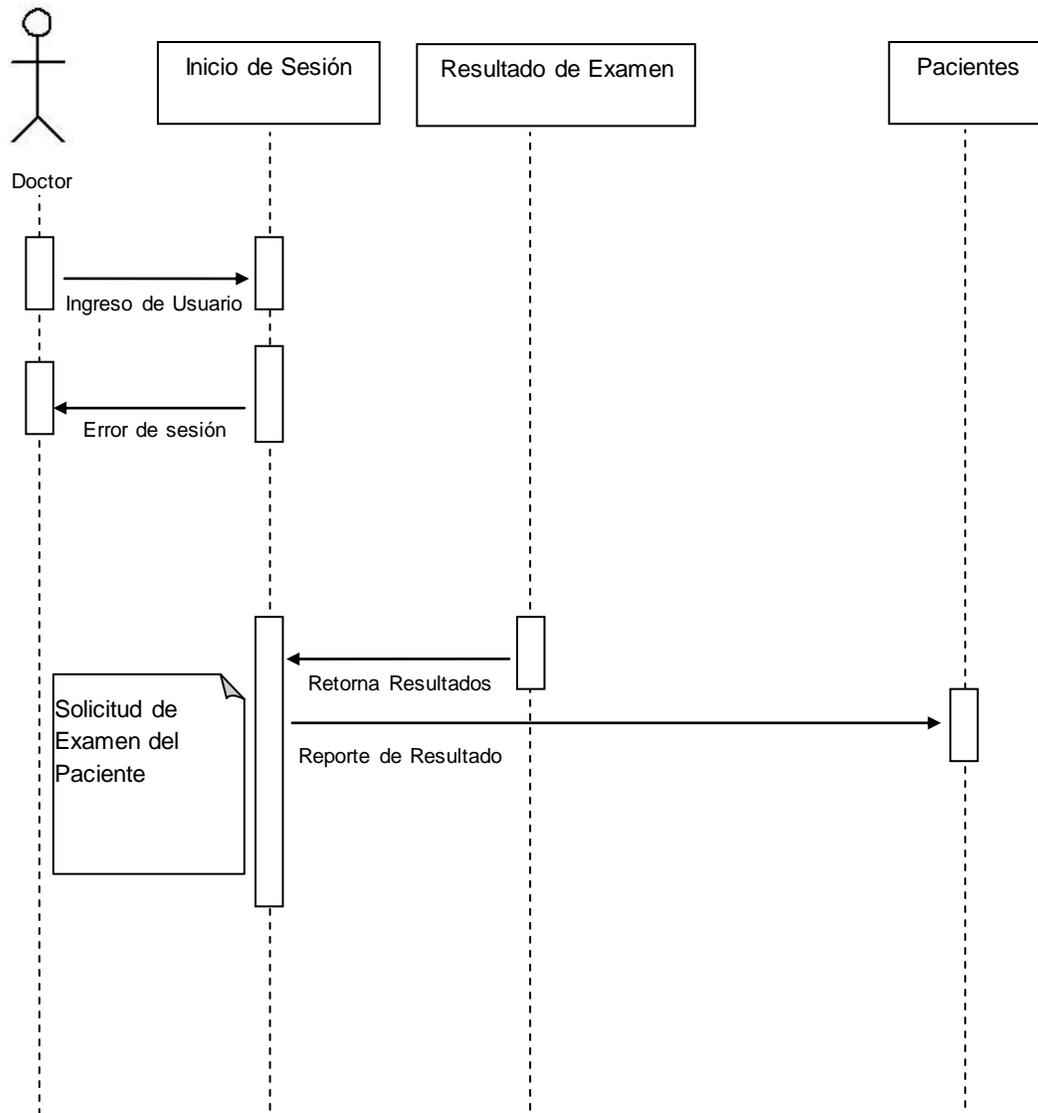


Grafico N° 20: Diagrama de secuencia

4.2.3 Diagramas De Actividad

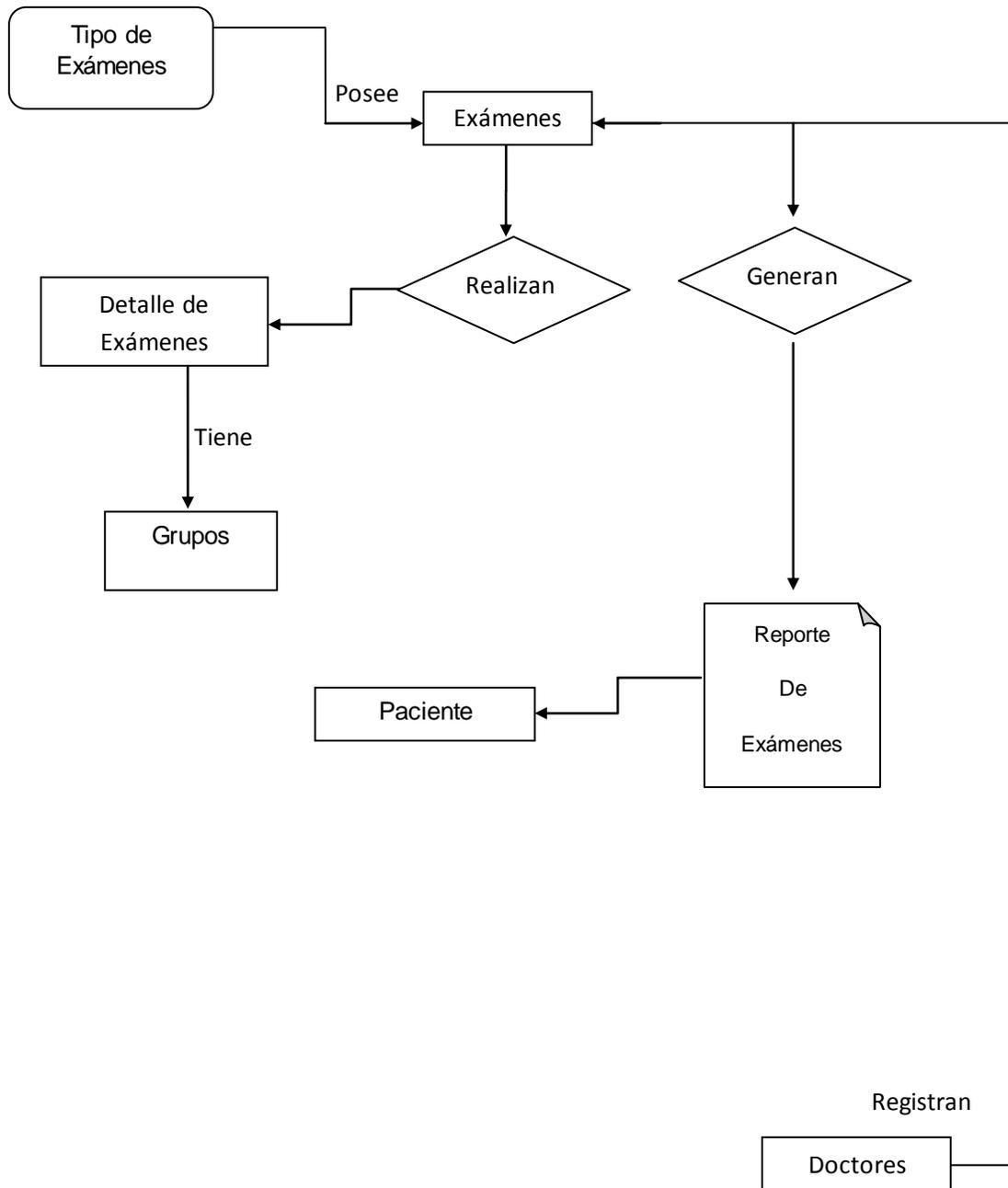


Grafico Nº 21: Diagramas de actividad

4.2.4 Diagramas De Despliegue

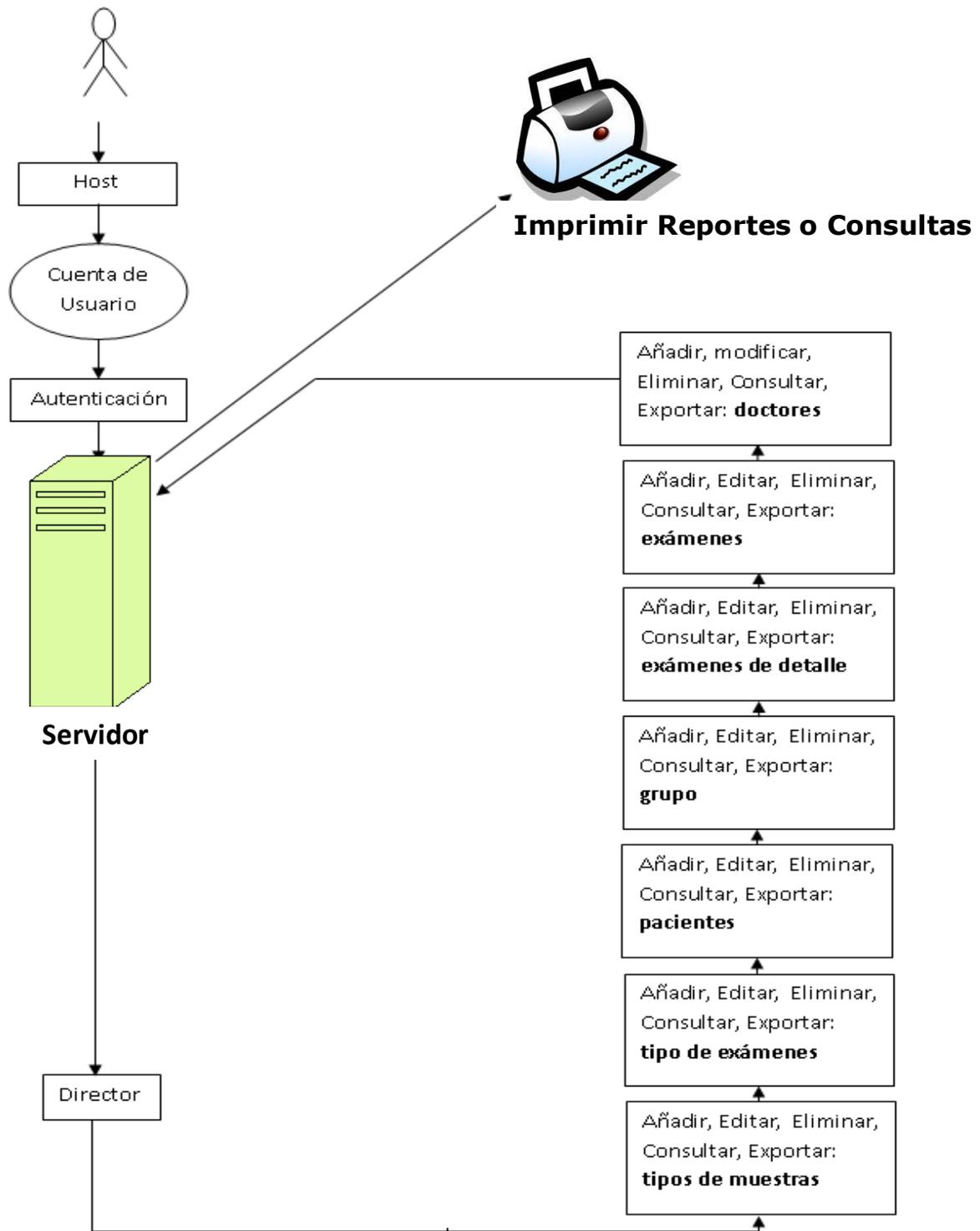


Grafico Nº 22: Diagramas de despliegue

4.2.5 Diagrama Entidad-Relación

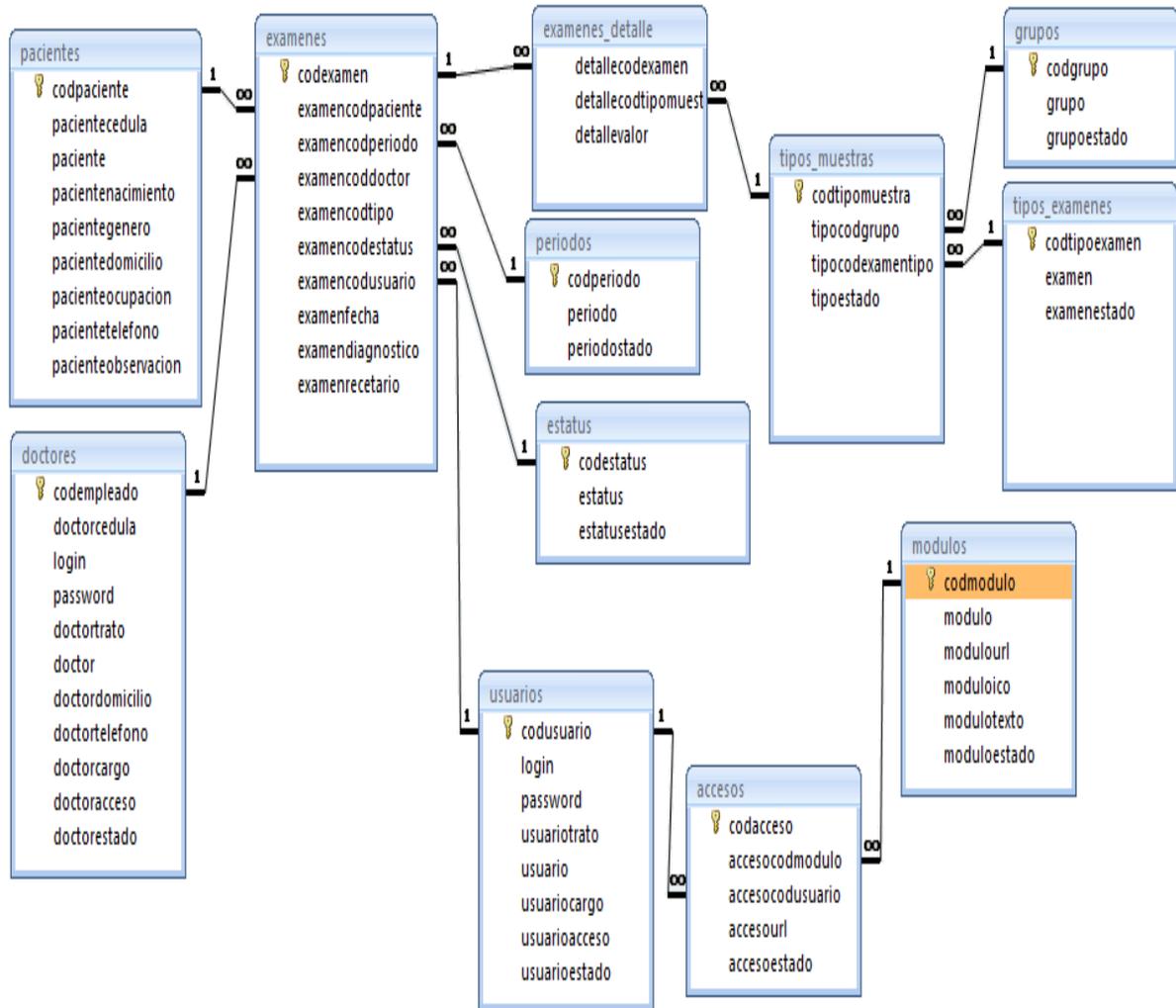


Tabla Nº 16: Entidad- Relación del sistema del laboratorio clínico

El **ER** representa las relaciones entre las tablas de la base de datos.

Cada tabla o entidad tiene sus propios atributos o campos.

Del lado de la aplicación, la persistencia mapea las tablas de la base de datos y crea las entidades, las cuales son objetos instanciables dentro de nuestro sistema.

4.2.6 Diccionarios De Datos

Diccionario

Procesos de Administrador

Nombre del Proceso	Descripción del Proceso	Datos de Ingreso	Datos de Salida
Módulos	En este proceso el administrador se encarga de realizar las diferentes cuentas para el ingreso del sistema.	Nombre de módulo; Pagina de enlace	Módulos de Usuario
Acceso	En este proceso el administrador se encarga de configurar los diferentes accesos al sistema dependiendo la clase de Modulo de Usuario	Módulo ; Usuario	Consultar Módulos
Usuarios	En este proceso se ingresa la información de los usuarios para que ingresen al sistema	Password ; Login ; Trato; Usuario; Cargo	Consultar Usuarios
Grupo	En este proceso se ingres el grupo de examen médico clínico	Grupo	Consultar Grupo
Tipos de Muestras	En este proceso se ingresa los tipos de exámenes con sus respectivas referencias medicas	Grupo ;Tipo de Examen ;Nombre de Muestra ; Siglas ;Referencia	Consultar Tipos de Muestras

Tabla Nº 17: Proceso de administrador

Proceso del Doctor

Nombre del Proceso	Descripción	Datos de Ingreso	Datos de Salida
Ingreso de Resultado de Examen Medico	En este proceso se ingresa la información del examen médico y sus resultados	N° Examen ;Paciente Validado por; Estatus ; Fecha de Registro ; Ingreso de datos de Examen	Reporte de Exámenes
Seleccionar Examen	En este proceso el doctor revisa los resultados del examen para poder dar su diagnostico del paciente que selecciona	Seleccionar Paciente; N° Examen ; Paciente ; Diagnóstico ; Receta	Reporte de Pacientes , Reporte de Exámenes

Tabla Nº 18: Proceso del doctor

Proceso de Enfermera

Nombre del Proceso	Descripción	Datos de Ingreso	Datos de Salida
Ingreso de Usuarios	En este proceso se ingresa la información personal del paciente o usuario	Cedula ; Paciente ;Ocupación Fecha de Nacimiento ;Domicilio Teléfono ;Genero Observación	Reporte de Paciente

Tabla Nº 19: Proceso de enfermera

Script.-Codificación de la base de datos.

A continuación exponemos el diccionario de datos y script de la base utilizada para la aplicación:

TABLA DIAGNOSTICOS

Consultar Diagnostico nos ayuda a encontrar el diagnostico de cada paciente de forma rápida y efectiva.

```
CREATE TABLE `diagnosticos` (  
  `coddiagnostico` varchar(3) NOT NULL,  
  `diagnosticocodtipomuestra` varchar(2) NOT NULL,  
  `max` varchar(8) NOT NULL,  
  `min` varchar(8) NOT NULL,  
  `diagnostico` varchar(200) NOT NULL,  
  PRIMARY KEY (`coddiagnostico`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

TABLA DOCTORES

Ahora se entrara como Usuario – Doctor el cual se desplegara el menú principal. Sistemas, Exámenes y Reportes.

Exámenes Médicos: aquí el doctor selecciona el paciente

```
CREATE TABLE `doctores` (  
  `coddoctor` varchar(3) NOT NULL,  
  `doctorcedula` varchar(10) NOT NULL,  
  `login` varchar(10) NOT NULL,  
  `password` varchar(10) NOT NULL,  
  `doctortrato` varchar(8) NOT NULL,
```

```
`doctor` varchar(50) NOT NULL,  
`doctordomicilio` varchar(50) NOT NULL,  
`doctortelefono` varchar(22) NOT NULL,  
`doctoracceso` varchar(8) NOT NULL,  
`doctorestado` varchar(8) NOT NULL,  
PRIMARY KEY (`coddoctor`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

TABLA ENTIDAD

```
CREATE TABLE `entidad` (  
  `entidad` varchar(200) NOT NULL,  
  `titulo` varchar(150) NOT NULL,  
  `direccion` varchar(150) NOT NULL,  
  `lugar` varchar(50) NOT NULL,  
  `telefono` varchar(20) NOT NULL,  
  `fax` varchar(20) NOT NULL,  
  `propietario` varchar(50) NOT NULL,  
  PRIMARY KEY (`entidad`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

TABLA ESTATUS

```
CREATE TABLE `estatus` (  
  `codestatus` varchar(2) NOT NULL,  
  `estatus` varchar(50) NOT NULL,  
  `estatusestado` varchar(8) NOT NULL,  
  PRIMARY KEY (`codestatus`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

TABLA EXAMENES

Exámenes Médicos: aquí el doctor selecciona el paciente

Una vez seleccionado el examen medico según el paciente, se da clic en **Seleccionar** el cual visualiza un informe.

```
CREATE TABLE `examenes` (  
  `codexamen` varchar(7) NOT NULL,  
  `examencodpaciente` varchar(6) NOT NULL,  
  `examencoddoctor` varchar(3) NOT NULL,  
  `examencodperiodo` varchar(2) NOT NULL,  
  `examencodtipo` varchar(2) NOT NULL,  
  `examencodestatus` varchar(2) NOT NULL,  
  `examencodusuario` varchar(2) NOT NULL,  
  `examenfecha` varchar(10) NOT NULL,  
  `examendiagnostico` varchar(300) NOT NULL,  
  `examenrecetario` varchar(300) NOT NULL,  
  PRIMARY KEY (`codexamen`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

TABLA EXAMENES_DETALLE

```
CREATE TABLE `examenes_detalle` (  
  `detallecodexamen` varchar(7) NOT NULL,  
  `detallecodtipomuestra` varchar(2) NOT NULL,  
  `detallevalor` varchar(7) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

TABLA GRUPOS

```
CREATE TABLE `grupos` (  
  `codgrupo` varchar(8) NOT NULL,  
  `grupo` varchar(50) NOT NULL,  
  `grupocolor` varchar(6) NOT NULL,  
  `grupoestado` varchar(8) NOT NULL,
```

```
PRIMARY KEY (`codgrupo`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

TABLA MODULOS

Módulos es aquel que se identifica como el ingreso del administrador del doctor para consultas.

```
CREATE TABLE `modulos` (  
  `codmodulo` varchar(2) NOT NULL,  
  `modulo` varchar(100) NOT NULL,  
  `modulourl` varchar(30) NOT NULL,  
  `moduloico` varchar(30) NOT NULL,  
  `modulotexto` varchar(150) NOT NULL,  
  `moduloestado` varchar(8) NOT NULL,  
  PRIMARY KEY (`codmodulo`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

TABLA PACIENTES

En la pestaña **Exámenes** se despliega la opción **Pacientes** en el cual vamos a ingresar los datos del paciente

```
CREATE TABLE `pacientes` (  
  `codpaciente` varchar(6) NOT NULL,  
  `pacientecedula` varchar(10) NOT NULL,  
  `paciente` varchar(50) NOT NULL,  
  `pacientenacimiento` varchar(10) NOT NULL,  
  `pacientegenero` varchar(9) NOT NULL,  
  `pacientedomicilio` varchar(50) NOT NULL,  
  `pacienteocupacion` varchar(50) NOT NULL,
```

```
`pacientetelefono` varchar(20) NOT NULL,  
`pacienteobservacion` varchar(200) NOT NULL,  
PRIMARY KEY (`codpaciente`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

TABLA PERIODOS

Periodos es una tabla que me permite ingresar un nuevo, guardar, y a la vez hacer una consulta del técnico en este caso el nombre en que la atendió del doctor en ese periodo.

```
CREATE TABLE `periodos` (  
  `codperiodo` varchar(2) NOT NULL,  
  `periodo` varchar(50) NOT NULL,  
  `periodoestado` varchar(8) NOT NULL,  
  PRIMARY KEY (`codperiodo`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

TABLA TIPOS_EXAMENES

Clic en **Resultados** y se despliega la ventana dónde el doctor emitirá su diagnostico medico

```
CREATE TABLE `tipos_examenes` (  
  `codtipoexamen` varchar(2) NOT NULL,  
  `examen` varchar(50) NOT NULL,  
  `examenestado` varchar(8) NOT NULL,  
  PRIMARY KEY (`codtipoexamen`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

TABLA TIPOS_MUESTRAS

```
CREATE TABLE `tipos_muestras` (  
  `codtipomuestra` varchar(2) NOT NULL,  
  `tipocodgrupo` varchar(2) NOT NULL,  
  `tipocodexamentipo` varchar(2) NOT NULL,
```

```

`tipomuestra` varchar(100) NOT NULL,
`tiposigla` varchar(15) NOT NULL,
`tiporeferencia` varchar(15) NOT NULL,
`tipoestado` varchar(8) NOT NULL,
PRIMARY KEY (`codtipomuestra`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1

```

TABLA USUARIOS

Se abrirá el menú del sistema en donde se muestra el menú y las diferentes clases de usuarios como Administrador, Usuario, Enfermera.

Al elegir cualquier clase de usuario se abrirá la ventana de seguridad la cual se repite en cualquier modulo del sistema. En este caso se inicia sesión como **Administrador**.

```

CREATE TABLE `usuarios` (
  `codusuario` varchar(4) NOT NULL,
  `usuariocedula` varchar(10) NOT NULL,
  `login` varchar(20) NOT NULL,
  `password` varchar(20) NOT NULL,
  `usuariotrato` varchar(8) NOT NULL,
  `usuario` varchar(50) NOT NULL,
  `usuariocargo` varchar(50) NOT NULL,
  `usuariodomicilio` varchar(50) NOT NULL,
  `usuariotelefono` varchar(20) NOT NULL,
  `usuarioacceso` varchar(8) NOT NULL,
  `usuarioestado` varchar(8) NOT NULL,
  PRIMARY KEY (`codusuario`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1

```

4.2.1 Procesamiento de la Información

La modalidad con la que se va a realizar la investigación del proyecto será básicamente de estas dos metodologías:

- **Cualitativa:** La **investigación cualitativa** o **metodología cualitativa** es un método de investigación usado principalmente en las ciencias sociales que se basa en cortes metodológicos basados en principios teóricos tales como la fenomenología, hermenéutica, la interacción social empleando métodos de recolección de datos que son no cuantitativos, con el propósito de explorar las relaciones sociales y describir la realidad tal como la experimentan los correspondientes. La investigación cualitativa requiere un profundo entendimiento del comportamiento humano y las razones que lo gobiernan. Se elige esta modalidad porque de esta forma determinaremos las características que contenga el problema planteado.
- **Cuantitativa:** La metodología cuantitativa es aquella que permite examinar los datos de manera científica, o de manera más específicamente en forma numérica, generalmente con ayuda de herramientas del campo de la estadística.

Con esta modalidad se podrá confirmar las características que contenga el problema ya planteado en relación a datos numéricos recolectados.

4.2.2 Análisis e Interpretación de Resultados

La encuesta realizada a los pacientes del laboratorio clínico del Centro de Salud de la ciudad de Echeandía

El sistema ideal de petición electrónica debería ayudar y guiar al clínico en el proceso de petición sugiriendo las pruebas a pedir en función del paciente y su proceso patológico, ofreciendo toda la información sobre las pruebas disponibles, avisando de posibles peticiones incongruentes o redundantes, informando de dónde y cuándo se puede obtener la muestra o si el paciente requiere una preparación especial.

Además, debería ser sencillo de manejar, ya que son miles las pruebas distintas que se pueden solicitar a un laboratorio. La sencillez se puede lograr mediante plantillas, perfiles o protocolos diagnósticos y un sistema eficiente de búsqueda de pruebas.

El diseño del sistema de petición va a tener un gran impacto en la demanda de pruebas de laboratorio.

El envío de muestras entre laboratorios es una práctica común. El SIL debería tener opciones que faciliten el envío y la gestión de muestras Manual de salud electrónica para directivos de servicios y sistemas de salud y datos para estos laboratorios. Lo ideal es que tanto el envío de datos como la recepción de resultados entre laboratorios sean electrónicos, lo que minimiza los errores y facilita el trabajo. En el caso de corporaciones o redes de laboratorios sería bueno que el SIL fuera multicéntrico, permitiendo una gestión corporativa de los laboratorios de la red. Es el análisis propiamente dicho. A partir de la muestra y la información se realizan las determinaciones correspondientes y se obtiene un resultado técnicamente correcto.

.4.3 Conclusiones y Recomendaciones

Conclusiones

- Durante el diagnóstico y análisis de la problemática del laboratorio de examen, nos hemos dado cuenta de que lleva la información a través de registros, donde debería contener un software en que me permita gestionar de manera rápida el ingreso de datos.

Por de un sistema nos permite optimizar el registro de manera rápida y eficaz, todos los registros de los pacientes, en donde que contenga una información de un resultado final.

Por medio de este sistema mejora el tiempo de respuesta de acuerdo a la necesidad del laboratorista como: guardar registró, modificar registró, elimina registró, consultas generales, consultas específicas, reportes.

- Una vez decididos, hay que tener en cuenta tanto lo que necesitamos, como lo que vamos a necesitar. Debe ser una solución que permita ampliar las funcionalidades que ofrece, modular a ser posible, que pueda crecer con nuestra empresa. No esperemos encontrar el producto perfecto. Necesariamente tendremos que cambiar la forma en la que hacemos las cosas para conseguir que el sistema funcione. Ninguno se va a adaptar perfectamente a nuestra manera de trabajar.
- Ya elegida, analicemos su compatibilidad. ¿Podemos aprovechar los datos de nuestro sistema actual? Debemos asegurarnos de que los podemos exportar al nuevo sin pérdidas de información. Casi todas las soluciones son compatibles con ciertos estándares, pero a la hora de la verdad, requiere bastante trabajo organizar los datos de forma que no se pierdan ni desordenen en el proceso.

4.4 Recomendaciones

- Al personal administrativo del laboratorio que aprovechen la informática para el ingreso de pacientes de la mejor forma.
- El personal administrativo del laboratorio tenga mejor prelación sobre informática a través de seminarios, capacitaciones y talleres etc.
- El director del Centro de Salud trate de hacer más utilidades con la informática, a fin de crear un sistema organizado a los pacientes.
- Para que la implementación del sistema sea 100% efectiva, es recomendable que se maneje con gran seguridad y reducir la cantidad de personas que utilicen el sistema para que no halle confusiones.
- Contar con un profesional independiente, que estudie tus necesidades, te ayude a elegir y supervise la implantación con el equipo de la solución escogida, es fundamental para garantizar el éxito en estos casos.

CAPITULO V

Propuesta

5.1 Título De La Propuesta

Mejoramiento del Centro de Salud de Echeandía.

5.2 OBJETIVOS

5.2.1 Objetivo General

Implementar una aplicación informática de arquitectura cliente-servidor que administre el laboratorio clínico del Centro de Salud mediante la aplicación PHP, con funcionalidades que sea de fácil entendimiento para el usuario, dando el asesoramiento técnico con Tutores especializados en la Base de Datos de MYSQL y SQLyog, Lenguaje de programación en PHP y Diseño de Sistemas Informáticos.

5.2.2. Objetivos Específicos

1. Elaborar el sistema informático que administre la consulta externa en el laboratorio clínico del Centro de Salud, mediante PHP y como base de datos MYSQL, que cumpla los requisitos necesarios para que se mejore su utilización en el cliente y servidor.
2. Realizar una investigación exhaustiva para aplicar los conocimientos adquiridos en la misma en el sistema informático que administre la consulta externa en el laboratorio.
3. Sintetizar el procedimiento y funcionamiento de la consulta externa en el laboratorio, para mostrarle al usuario en la aplicación, lo más trascendental del tema, plasmado de la misma.

5.3 Justificación

En la presente investigación y análisis está encaminada para mejorar el Centro de Salud de Echeandía ante la comunidad, proporcionando las soluciones a la problemática que enfrenta dicha Institución pública de salud.

En el Centro de Salud especialmente el laboratorio hay pérdidas y anomalías que han ocurrido a través del ingreso manual de cada registro de los pacientes y la incomodidad que causa al paciente el trámite del papeleo ; y el tiempo que tarda, en ingresar los datos de manera manualmente por cada paciente.

Por eso se a decidido implementar un sistema informático que tenga un ingreso y a la vez acceso rápido a la información y por ende mejorar en la atención a los usuarios.

Mayor motivación en los mandos medios para anticipar los requerimientos de las directivas. Generación de informes e indicadores, que permiten corregir fallas difíciles de detectar y controlar con un sistema manual. Aumento de la productividad gracias a la liberación de tiempos en búsqueda y generación de información repetida.

El sistema será una solución en informática que permitirá controlar la administración y las operaciones de su laboratorio de manera ágil, amigable y confiable.

Sus objetivos principales son: mejorar la imagen y el control en el proceso de resultados.

El sistema tendrá en cuenta con las funciones adecuadas para medir la productividad y crecimiento de su Laboratorio del Centro de Salud.

En el análisis profundo realizado en el centro de salud del laboratorio se ha dado a conocer la falta de un sistema informático en que administre el ordenamiento y la agilidad en lo que es los ingresos de datos o información, ya que es muy ventajosa la gran seguridad que tendrá este sistema web, que se aplicara en PHP RUNER y como motor de base de datos se lo realizara en SQLyog como generador de base de datos, donde contiene una mayor seguridad para el usuario y fácil de utilizarlo en el manejo del diseño del sistemas.

5.4 Importancia

La importancia del PHP en la cotidianidad de las páginas web, y en el contacto con el usuario, permitiendo a los desarrolladores interactuar con estos a través de las páginas dinámicas.

La importancia del lenguaje de programación PHP Runner es un excelente programa para acceder y modificar bases de datos de MySQL y SQLyog. Se ingresa por medio de una cuenta de usuario y permite así generar páginas de PHP en la que puedes buscar, clasificar, corregir y suprimir en la base de datos.

El programa genera códigos puros de PHP. Las páginas que se crean pueden ser protegidas por medio de contraseñas de PHP. Es un simple pero útil programa para pre visualizar fácilmente un archivo PHP sin necesidad de grabarlo o subirlo a un servidor. Dispone de una interfaz amigable con herramientas útiles como botones para las funciones PHP más habituales, para hacer el script sencillo y rápido.

5.5 Viabilidad

5.5.1 Viabilidad Técnica

El sistema realizado será viable ya que el GRUPO MÉDICO ESPECIALIZADO cuenta con los elementos necesarios para la implantación del programa, aunque tendremos que asesorar al personal como utilizar el sistema para que la viabilidad no sea limitada.

5.5.2 Viabilidad Económica

Si los costos son menores que los beneficios hay viabilidad económica ya que los sistemas de software libres existentes permiten la instalación del software tantas veces y en tantas máquinas como el usuario desee. Uno de los modelos de negocio que genera el software libre es la contratación de servicios de atención al cliente. Este sistema permite que las compañías que den el servicio compitan en igualdad de condiciones al no poseer la propiedad del producto del cual dan el servicio.

5.5.3 Viabilidad Operativa

El programa podrá ser manejado por el mismo personal:

- Al ingresar al sistema se podrá dar cambios de usuario y password.
- La ejecución del sistema en un equipo recomendado será más rápido.
- Los pacientes tendrán códigos, el laboratorista agilizará el manejo de los datos.
- Tendrá mayor seguridad ya que solo puede acceder al sistema los doctores y el laboratorista ó (secretaria).

Como podemos ver los cambios no serán drásticos, por ende si hay viabilidad operativa.

5.6 Fundamentación Teórica (Propuesta)

El Laboratorio Clínico es una herramienta primordial para el área médica, ya que por medio de este se diagnostican diferentes patologías y además se realizan estudios para establecer el tipo de tratamiento que se debe administrar al paciente, al igual que el seguimiento del mismo.

El programa cuenta con tres hitos principales: obtener un proceso definido, obtener un proceso validado y obtener un proceso ajustado y mejorado. Los hitos debe ser detallado

En el presente trabajo, existe el compromiso de investigar y conocer de manera práctica la aplicabilidad de programar en modo gráfico, utilizando la tecnología de PHP y MySQL, en un código que resuelva la administración de consultas generales y consultas específicas en el laboratorio clínico de un Centro de Salud Público.

Entre los temas a estudiar serán:

1. Programación en PHP y HTML.
2. Principales temas de consulta externa de acuerdo a las necesidades del laboratorista.

3. Principales temas de Bases de datos de **MYSQL** y **SQLyog**.

La aplicación en PHP, que administrará la consulta externa y tendrá una arquitectura cliente-servidor, en cual, tendrá su conexión con la base de datos correspondientes, la misma que servirá para imprimir reportes.

Por medio de este sistema web tendría mucha confiabilidad y seguridad al ser almacenado en la base de datos, ya que por ende no habría pérdida de datos.

5.7 Descripción de la Propuesta

- Consulta de resultados en línea.
- Mejor imagen en los resultados de los estudios.

Esta investigación realizada en el Centro de Salud de la ciudad Echeandía prov. Bolívar, ubicada en la avenida 5 DE OCTUBRE junto al estadio monumental, donde cuenta con dos equipos necesarios para ser utilizada como manejo del sistemas, la cual consta con las características necesarias, donde el personal para el ingreso de datos se realiza a través de registros, la cual existe mucha anomalía, a la implementación de este sistema daría una mejor satisfacción para el laboratorista y a la población aledaña.

Mediante este sistema tendrá mayor capacidad de atención en los pacientes al ser recibido los registros realizado los análisis del laboratorio como una impresión de resultado final.

5.8 Ejecución de la Propuesta

Estrategias de Desarrollo

La **aplicación** de este sistema web se lo va desarrollar en PHP, y HTML utilizando MYSQL como base de datos.

Para el funcionamiento de este sistema tendrán las siguientes opciones:

- ❖ Ingreso al usuario.
- ❖ Creación de nuevo usuario
- ❖ Página principal con sus respectivos logos.
- ❖ Ingresos de datos de pacientes
- ❖ Tipos de exámenes
- ❖ Consultas generales
- ❖ Consultas específicas
- ❖ Reportes
- ❖ Guardar, eliminar, modificar registros.

5.9 Presupuesto

Presupuesto/Recursos	Cantidad	Costo Unitario	Costo Total
Horas de uso de internet	80	1.00	80.00
Impresión de documentación	150	0.20	30.00
Suministro y materiales	2	16.00	32.00
Mantenimiento del software	1	300.00	300.00
Costo del software	1	1500.00	1500.00
Transporte	20	4.00	80.00
Diseño de la página Web	30	7.00	210.00
Equipo (computadora)	1	980.00	980.00
Software en SO original	1	130.00	130.00
Total			3342.00

Tabla Nº 20: Presupuesto

5.10 Cronograma Tentativo (Ejecución Propuesta)

Tareas / Actividades	M 9	M 10	M 11	M 12	M 1	M 2	M 3	M 4	M 5	M 6	M 7	M 8	M 9
Realización del anteproyecto	█												
Aprobación del anteproyecto		█											
Realización del proyecto			█										
Creación base de datos				█									
Desarrollo de sistema				█									
Diseño de pantalla			█										
Pantalla de usuario				█									
Pantalla de entrada						█							
Pantalla de salida								█					
Validaciones										█			
Pruebas y depuración de Errores del sistema											█		
Defensas de tesis													█

Tabla N° 21: Cronograma tentativo

5.11 Evaluación

La revisión y la evaluación de los controles, sistemas, procedimientos de informática; de los equipos de cómputo, su utilización, eficiencia y seguridad, de la organización que participan en el procesamiento de la información, a fin de que por medio del señalamiento de cursos alternativos se logre una utilización más eficiente y segura de la información que servirá para un mejor resultados en la impresión para el paciente.

5.12 Impacto

El software que se va a desarrollar consta con tecnología Php debido las razones de utilizar este lenguaje se deben a su poder y sencillez. PHP bien es un software libre, no es necesario pagar para poder utilizarlo (como lo es con asp) y una de sus grandes cualidades es su versatilidad al momento de escribir código, su sencillez en la sintaxis, e inclusive su seguridad.

Mediante la integración de estas herramientas tecnológicas al Laboratorio se podrá brindar un servicio rápido, ágil y efectivo el cual beneficiará tanto a los clientes en este caso serán los pacientes y también a los que administraran esté software debido a que se registrara y se dará información d una manera eficaz. Brindando los siguientes beneficios.

5.12.1 Beneficios del Sistema:

- Registro automatizado y manual de resultados.
- Impresión y validación de reportes de laboratorio.
- Almacenamiento de la información histórica de los pacientes y sus resultados.

5.12.2 Económico:

- No tiene que comprar un sistema caro para el laboratorio.
- Es un sistema donde el cual se usa herramientas muy económicas respecto a su gran funcionamiento.

5.12.3 Seguro:

- Sistema protege acceso a información con claves y perfiles de seguridad.
- Sistema y programa son respaldados diariamente.
- Interfaces con equipos de laboratorio automatizados.
- Se garantiza confidencialidad de la información.

El centro médico de Echeandía genera expectativa d atención a la ciudadanía indiscutiblemente con su mejoramiento, eso incidirá en una mejor atención a la población y el impacto social será mayor cobertura.

En el centro de salud de Echeandía tiene gran acogida de pacientes a diario en la cual, con la implementación de este sistema web mejora la agilidad de ingresos de datos.

Conclusiones

- ❖ Como conclusión de este sistema realizado en la aplicación del lenguaje de programación PHP, diseñado originalmente para la creación de páginas web dinámicas y con conexión de base de datos SQLYog, ya que es un administrador de base de datos MySQL para Windows que, recientemente, ha liberado su versión para código abierto, gratuita para el uso no comercial.
- ❖ El PHP y el MySQL son herramientas poderosas para la elaboración de aplicaciones web dinámicas.
- ❖ Este sistema brindara rapidez y agilidad al momento de ingresar datos de los pacientes en sus fichas médicas virtuales, así mismo proveerá los informes a los pacientes de manera ordenada agilizando el proceso de respuesta.
- ❖ Además este sistema se podrá dar cambios o modificados los usuarios como servidor para el cliente de acuerdo a la categoría y su seguridad de su password para ingreso del sistema con las personas indicadas.

Recomendaciones

- ❖ Al personal administrativo del laboratorio se recomienda de que tenga la orientación necesaria para el manejo del sistema web de la mejor forma.
- ❖ El personal administrativo del laboratorio debe recibir seminario sobre la informática a través de los seminarios, capacitaciones y talleres etc.
- ❖ El director del Centro de Salud trate de hacer más útil y eficiencia con la informática, a fin de crear un sistema web muy organizado para el ingreso de los pacientes de forma rápida y eficaz.
- ❖ Contar con un profesional capacitado, en caso de llegarse a dañar este sistema, tenga la solución o el mantenimiento necesario para el sistema web.
- ❖ Como recomendación de este sistema web se debe obtener los siguientes requisitos: Plataforma, xp2, xp3, Windows 7 y Linux. de acuerdo a la característica con una capacidad adecuada.
- Implementar en el computador proxy (comparte el servicio de Internet) el sistema operativo Linux, a fin de aminorar el filtrado de virus informáticos.
- El manejo en modo administrador deber realizado cuidadosamente.
- Las claves de los usuarios internos deberán ser actualizadas cada tres meses.

Glosario de Términos y Abreviaturas

ATRIBUTO: un atributo es una especificación que define una propiedad de un Objeto, elemento o archivo, también puede referirse o establecer el valor específico para una instancia determinada de los mismos, ya que esta denominado a un conjunto de reglas.

APACHE: es usado principalmente para enviar páginas web estáticas y dinámicas en la World Wide Web, muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o que utilizarán características propias de este servidor web.

CUALITATIVA: La investigación cualitativa requiere un profundo entendimiento del comportamiento humano y las razones que lo gobiernan.

CUANTITATIVA: es aquella que permite examinar los datos de manera científica, o de manera más específicamente en forma numérica, generalmente con ayuda de herramientas del campo de la estadística.

COOKIES: Para PHP las cookies son fáciles de utilizar. Para esto se utiliza la función `setcookie()` y al igual que la función anterior, `Header`, tiene que ser invocada antes de producir ninguna salida.

ES LIBRE: por lo que se presenta como una alternativa de fácil acceso para todos, permite aplicar técnicas de programación orientada a objetos.

ENCAPSULACIÓN: Propiedad que permite ocultar la información al resto de los objetos, impidiendo así accesos incorrectos o conflictos.

El servidor HTTP Apache: es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1² y la noción de sitio virtual.

GPL: es la *Licencia Pública General de GNU* (GNU GPL), ó La Licencia Pública General de Affero AGPL, es una licencia copyleft derivada de la Licencia Pública

General de GNU diseñada específicamente para asegurar la cooperación con la comunidad en el caso de software que corra en servidores de red.

HTML: es el acrónimo inglés de **HyperText Markup Language**, que se traduce al español como *Lenguaje de Etiquetas de Hipertexto*, es un lenguaje de marcado diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web.

HERENCIA: Propiedad a través de la cual los objetos heredan comportamiento dentro de una jerarquía de clases.

JERARQUÍA: es el criterio que permite establecer un orden de superioridad o de subordinación entre personajes, instituciones o conceptos, los sistemas jerárquicos impregnan la vida cotidiana por ejemplo, el ejército, que cuenta con los generales.

LICENCIA DE SOFTWARE: Una **licencia de software** es un contrato entre el licenciante (autor/titular de los derechos de explotación/distribuidor).

MySQL: es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones, es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. MySQL AB fue fundado por David Axmark, Allan Larsson y Michael Widenius.

OPEN SOURCE. El código fuente de MySQL se puede descargar y está accesible a cualquiera, por otra parte, usa la licencia GPL para aplicaciones no comerciales.

PHP Runner.- construye visualmente atractiva interfaz web para cualquier local o remota MySQL, MS Access, SQL Server y bases de datos Oracle.

PHP: (acrónimo de *PHP: Hypertext Preprocessor*) es un lenguaje de código abierto muy popular especialmente adecuado para desarrollo web y que puede ser incrustado en HTML, puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes Unix.

POLIMORFISMO: Propiedad de una operación mediante la cual puede ser aplicada a distintos tipos de objetos.

PROLOG: prolog, es un lenguaje de programación lógica es conjunto de hechos y reglas que representan el problema que se pretende resolver.

SGBD: Los sistemas de gestión de bases de datos (en *database management system*, abreviado *DBMS*) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

SOFTWARE LIBRE.- Software Libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software.

SISTEMAS INFORMÁTICOS.- Informática proviene de la fusión de los términos “información” y “automática”, y se define como la ciencia que estudia el tratamiento automático.

SISTEMAS NO LIBRES: como por ejemplo MySQL Licencia Dual, depende del uso no se sabe hasta cuándo permanecerá así, ya que ha sido comprada por Oracle.

URL: significan Universal Resource Locator (Localizador Universal de Recurso). Esta interpretación puede ser debida al hecho de que, aunque la U en URL siempre ha significado Uniforme, en la World Wide Web (WWW o Web). Desde 1994, en los estándares del internet, el concepto de URL ha sido incorporado dentro del más general de URI (Uniform Resource Identifier - Identificador Uniforme de Recurso), pero el término URL aún se utiliza ampliamente.

Referencias Bibliográficas

Autor: Jorge Sánchez (www.jorgesanchez.net) año 2004

Copyright © 2001-2011 The PHP Group , <http://php.net/manual/es/intro-whatism.php>,

<http://es.answers.yahoo.com/question/index?qid=20070907104610AAxlfH4>

<http://es.wikipedia.org>

http://es.wikipedia.org/wiki/Base_de_datos#Modelos_de_bases_de_datos

<http://es.wikipedia.org/wiki/HTML>

<http://es.wikipedia.org/wiki/HTML>

http://es.wikipedia.org/wiki/Licencia_de_software<http://www.uaem.mx/posgrado/mcruz/cursos/miic/MySQL.pdf>

<http://php.net/manual/es/intro-whatism.php>

<http://www.monografias.com/trabajos28/informatica-hospital/informatica-hospital.shtml>

<http://www.mysql.com>

<http://www.opencontent.org/openpub/>

<http://www.php.net/manual/es/install.php>

<http://www.php.net/manual/es/intro-whatcando.php>

<http://www.uaem.mx/posgrado/mcruz/cursos/miic/MySQL.pdf>



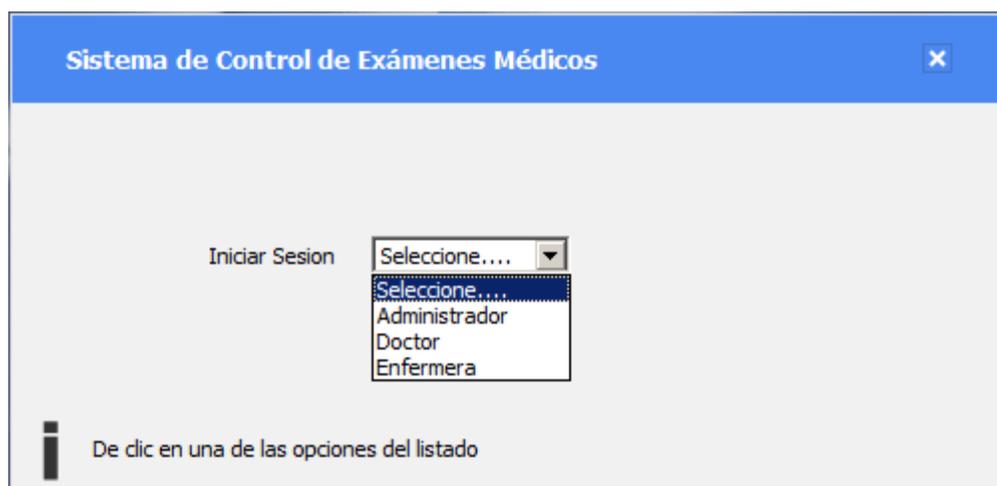
Manual de Usuario

Como Ingresar al Sistema

1

Para ingresar a la página de Internet, abrimos **Internet Explorer** o cualquier otro navegador y escribimos en la barra de dirección:

http://localhost/sistema_examenes_medicos/

**2**

Ingresamos a los perfiles Administrador, Doctor y Enfermera el cual permitirá accesos segun su perfil el cual detallamos a continuación:

Doctor.- Encargado de diagnosticar el tratamiento y enfermedad de los pacientes

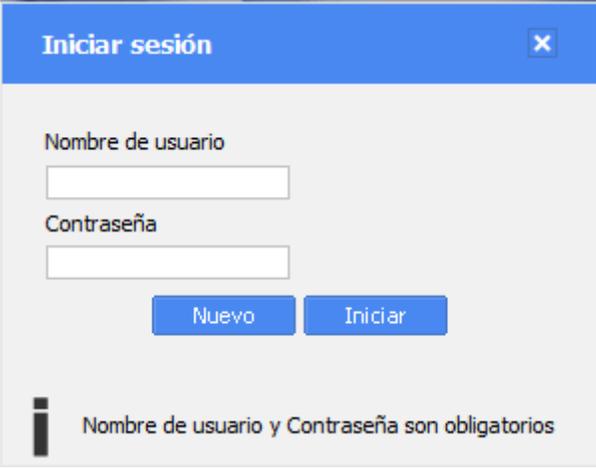
Administrador.- Este usuario permite configurar y agregar todos los datos del los exámenes médicos.

Enfermera.- Ingresa a os pacientes y los valores de los exámenes médicos

3

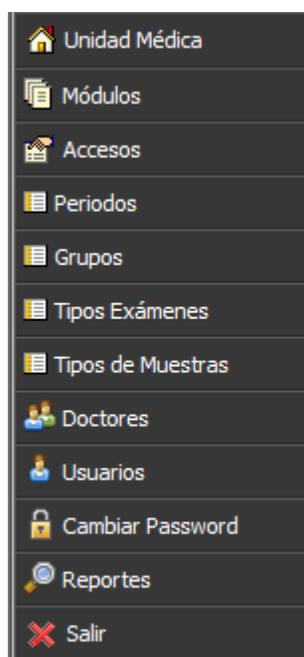
Accedemos al sistema ingresando el Login (Nombre de Usuario) y el Password (Contraseña) asignada.

Veremos la siguiente pantalla:



The screenshot shows a login window titled "Iniciar sesión" with a close button (X) in the top right corner. The window contains two input fields: "Nombre de usuario" and "Contraseña". Below the input fields are two buttons: "Nuevo" and "Iniciar". At the bottom left, there is an information icon (i) followed by the text "Nombre de usuario y Contraseña son obligatorios".

El manejo de los las diferentes opciones de la barra de menú se detalla a continuación:



4

Unidad Médica.- Almacena toda la información referente al lugar donde se va a implementar el software.

Módulos Asiente para agregar tipos de perfil y permite futuras ex paciones del Sistema permitiendo administrar otras áreas.

Nombre de Modulo	Nombre o Título del modulo
Pagina de enlace	Nombre de la página administrativa del modulo
Estado	Habilita de deshabilita sin eliminar (Activo/Inactivo)

Accesos Establece el enlace de accesos de los usuarios a los diferentes Módulos

Módulo	Selecciona el módulo al que deseamos realizar el acceso
Usuario	Agrega el usuario que se desee que ingrese al módulos
Estado	Habilita de deshabilita sin eliminar (Activo/Inactivo)

Periodos

Para crear un nuevo Periodo realice lo siguiente:

1. A continuación seleccione la opción Periodos

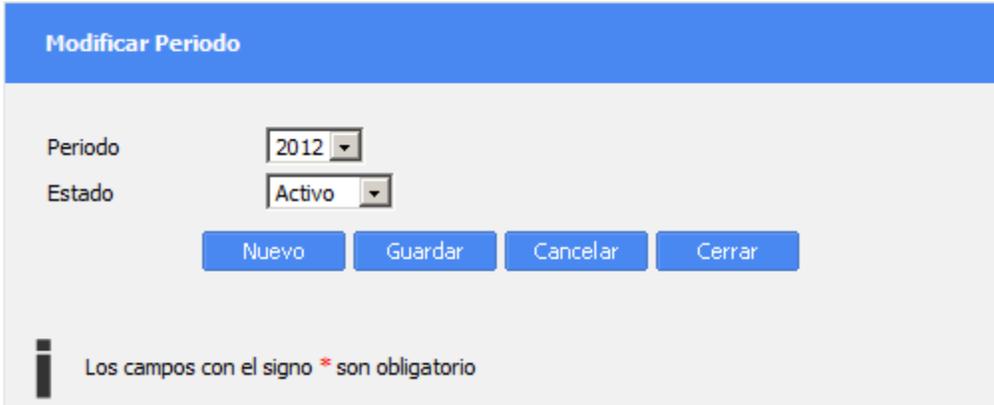
2. Presione clic en el botón  para agregar un nuevo registro en la base de datos.

Modificar Periodo

Algunas veces se comete errores ortográficos en los campos de las Periodos o se desea cambiarlo, en estos casos es muy útil la opción *Modificar Periodo*. Para Editar

realice lo siguiente:

1. Presione clic en el botón 
2. A continuación le mostrar la siguiente ventana
3. Luego presione el botón buscar



Modificar Periodo

Periodo

Estado

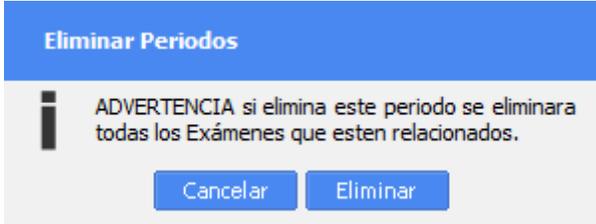
 Los campos con el signo * son obligatorio

4. Parar guardar de clic en el botón 

Eliminar Periodo

Para eliminar un Periodo de sistema realice lo siguiente:

1. Seleccione el Periodo que desea eliminar
2. Le mostrar un mensaje de Advertencia



Eliminar Periodos

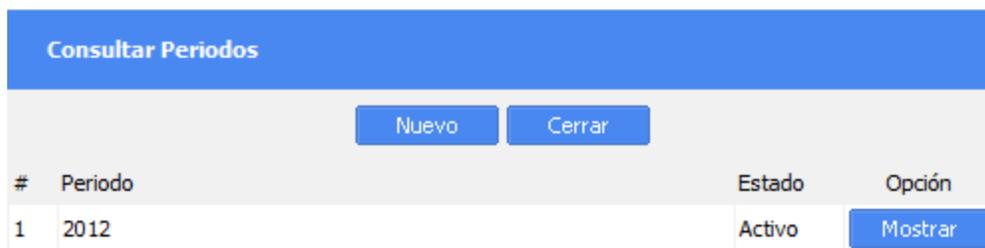
 ADVERTENCIA si elimina este periodo se eliminara todas los Exámenes que esten relacionados.

3. Presione clic en el botón  si desea eliminar
4. en caso contrario 

Nota: No es recomendable **Eliminar Periodos** por que se encuentran relacionados con las tablas Exámenes Médicos.

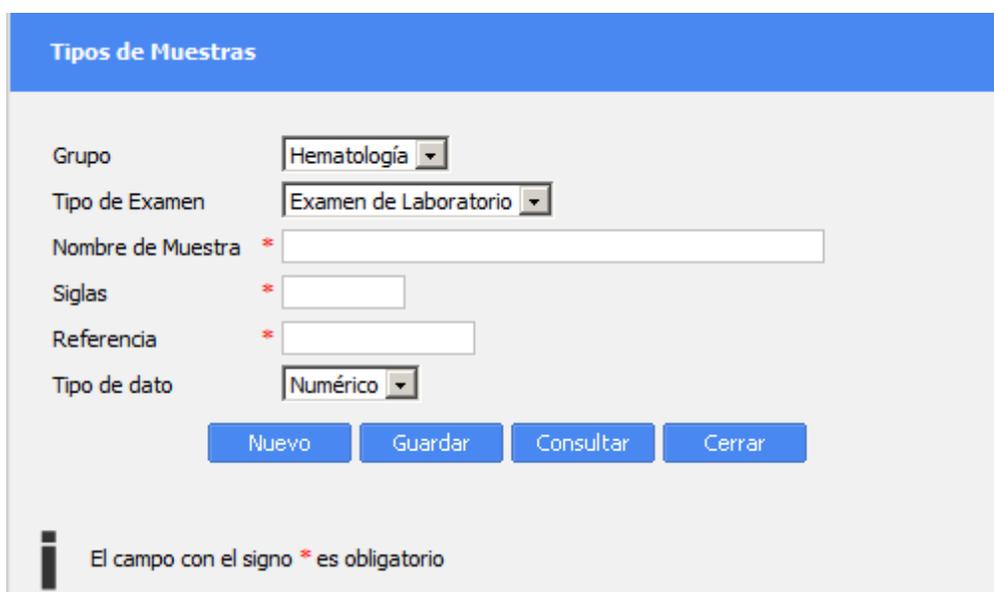
Consultar Periodos

Para consultar las Periodos de clic en el botón 



#	Periodo	Estado	Opción
1	2012	Activo	

Grupos define los lugares o espacios donde labores los Empleados de la Empres



Grupos	Nombre del Grupo
Tipo de Examen	Tipo de Examen
Nombre de Muestra	Nombre de Muestra
Sigla	Iniciales del la Muestra
Referencia	Valores de Referencia
Tipo de Datos	Establece que validación se realiza según el tipo de datos
Estado	Habilita de deshabilita sin eliminar (Activo/Inactivo)

Usuarios Esta opción permite ingresar al personal encargado de la Administración y Funcionamiento del Sistema así como también mantener la seguridad de accesos

Password	Una contraseña o clave autenticación que utiliza información secreta para controlar el acceso
Login	Es un Alias del Usuario al momento de autenticación al ingresar a un servicio o sistema.
Usuario	Nombre del Usuario que tendrá acceso al sistema
Trato	Manera de proceder del Usuario en relación con los demás
Cargo	Establece el nivel de acceso al sistema
Estado	Habilita de deshabilita sin eliminar (Activo/Inactivo)

Doctor Persona encargada de realizar el diagnostico del paciente

Cedula	Identificador único numérico
Password	Una contraseña o clave autenticación que utiliza información secreta para controlar el acceso
Login	Es un Alias del Usuario al momento de autenticación al ingresar a un servicio o sistema.

Doctor	Nombre del Usuario que tendrá acceso al sistema
Trato	Manera de proceder del Usuario en relación con los demás
Domicilio	Lugar donde reside el doctor
Teléfono	Número telefónico personal del doctor
Estado	Habilita de deshabilita un usuario sin eliminar (Activo/Inactivo)

The screenshot shows a web form titled "Doctores" with a blue header. Below the header, there are seven input fields, each with a label and an asterisk indicating it is mandatory: "Cedula *", "Password *", "Login *", "Trato *", "Doctor *", "Domicilio", and "Teléfono". Below the fields are four blue buttons: "Nuevo", "Guardar", "Consultar", and "Cerrar". At the bottom left, there is an information icon and the text "Los campos con el signo * son obligatorio".

5

Cambiar Password.- Este formulario permite modificar el Login y el Password y actualizar sus datos si el Docente no posee cuenta de usuario, deberá solicitar en secretaria la creación es de inmediato.

Al seleccionar esta opción les mostrara la siguiente pantalla:

Cuenta de Usuario

Password * ●●●●●

Login * admin

Trato Ing.

Usuario Freddy Muñoz

Cargo Técnico

Guardar Cerrar

Los campos con el signo * son obligatorio

6

Cerrar Sesión.- Esta opción de terminar una sesión de un usuario específico, finalizando así el uso del sistema.

Cerrar Sesión

¿Está seguro de salir del sistema?

No Si

7

Reportes. Listado con formato de documento A4 el cual puede ser impreso en cualquier momento.



LABORATORIO CLINICO ECHEANDIA

Sistema de Control de Exámenes Médicos

Babahoyo - Ecuador

Direc.: Av. Angel Custodio Sanchez 312 y Flores

Telé.: 2737097 - 092370912

Reporte de Pacientes

« 1

1 »

Imprimir

Nro.	Paciente	Domicilio	Teléfono
1	ABAD MONTENEGRO MARIA TERESA	CDLA. OTTO AROSEMENA	094058364
2	ABAD SUAREZ BRAYAN ALEXIS	CDLA. LA VENTURA	
3	ABRIL ARZUBE SERGIO ANDRES	BY PASS	
4	ACOSTA BAQUERIZO MILEINER EVELYN	BARREIRO	
5	ACOSTA CASTRO HENRY STANLY	BARREIRO	
6	ACOSTA CUZCO MARCOS BIKEISY	CALLE GUARANDA Y LA 9NA. BARREIRO	094058364
7	ACOSTA DIAZ JEAN PAUL	ROCAFUERTE Y CUSTODIO SANCHEZ	
8	ACOSTA MACIAS MARIA JOSE	EL SALTO	086425789
9	ACOSTA MARTINEZ DAVID FERNANDO	EL SALTO	093647336

1/1



LABORATORIO CLINICO ECHEANDIA

Sistema de Control de Exámenes Médicos

Babahoyo - Ecuador

Direc.: Av. Angel Custodio Sanchez 312 y Flores

Telé.: 2737097 - 092370912

Reporte de Exámenes

« 1

1 »

Imprimir

Nro.	Nº Exm.	Paciente	Tipo de examen	Fecha
1	0000001	ABRIL ARZUBE SERGIO ANDRES	Examen de Laboratorio	19/07/2012
2	0000002	ACOSTA CASTRO HENRY STANLY	Examen de Laboratorio	19/07/2012
3	0000003	ACOSTA DIAZ JEAN PAUL	Examen de Laboratorio	30/08/2012
4	0000004	ABAD MONTENEGRO MARIA TERESA	Examen de Laboratorio	30/08/2012

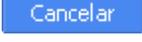
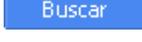
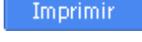
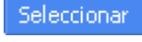
1/1

Botones de Navegación

«	Botón Inicio.- Muestra los registro del la primera hoja
»	Botón Siguiente.- Muestra los registros de la siguiente pagina

	Botón Anterior.- Muestra los registros de la página anterior
	Botón Ultimo.- Muestra los registros de la ultima hoja

Botones de los Formularios

	Botón Nuevo Crea un nuevo registro en la base de datos
	Botón Guardar Almacena un nuevo registro o actualiza uno ya existente
	Botón Modificar Permite Editar un registro ya existente.
	Botón Eliminar Borra un registro de la base de datos
	Botón Cancelar Permite anular una orden de nuevo o modificar
	Botón Consultar muestra el listado de los registro almacenados
	Botón Buscar examina un registro
	Botón Imprimir Permite imprimir un lista o reporte
	Botón Seleccionar Permite elegir un registro de un listado relacionado

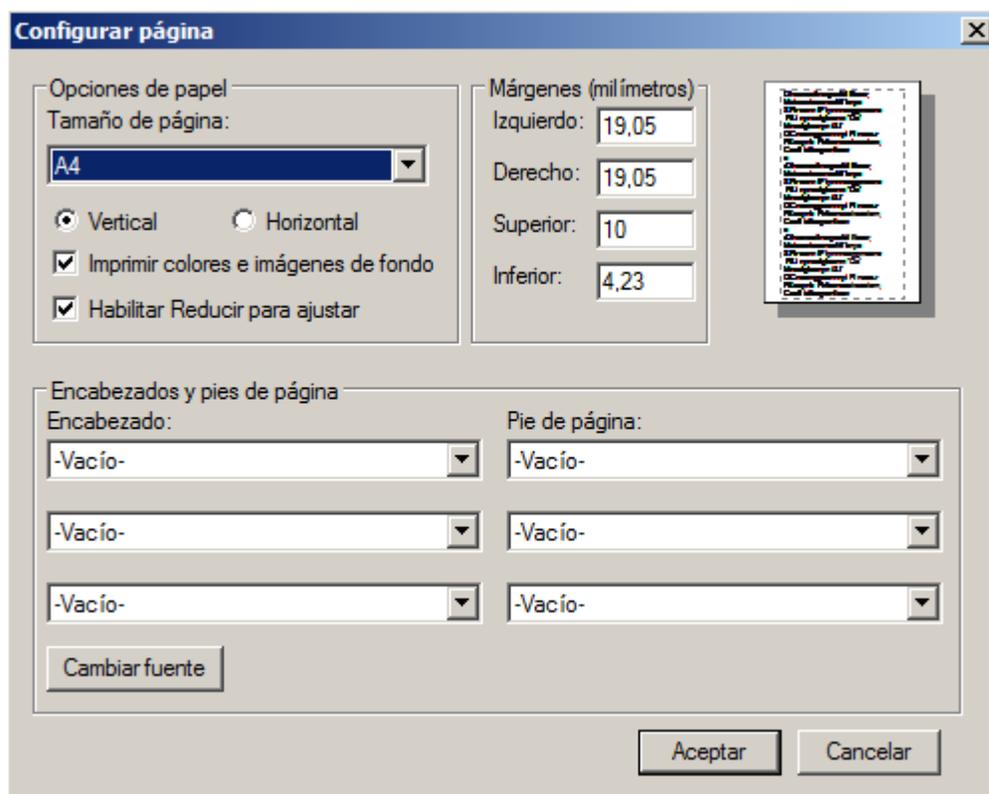
8

Configuración de Impresión

En el navegador Internet Explorer en la barra de menú seleccionamos **Archivo** y luego la opción **Configurara página** le mostrara la siguiente ventana.

Para la correcta impresión de los reportes la configuración de su navegador debe de ser igual a esta ventana

Navegador Internet Explorer



Para la correcta impresión de los reportes la configuración de su navegador debe de ser igual a esta ventana

Navegador Mozilla Firefox

