



Título del Proyecto

**“IMPLANTACIÓN DEL SISTEMA DE EVALUACIÓN CURRICULAR
PARA EL BACHILLERATO INFORMÁTICO DEL COLEGIO SULTANA
DEL ORIENTE DE LA CIUDAD DE MACAS EN ENTORNO WEB,
APLICANDO LA NORMA ISO/IEC 9216”**

Autores:

Baldeón López Marco Aníbal
Narváez Gavilanes Carlos Fabricio

Director:

Ing. Danny Velasco MsC

UNIVERSIDAD NACIONAL DE CHIMBORAZO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERIA EN SISTEMAS Y COMPUTACIÓN

Riobamba – Ecuador

AÑO

2012

CALIFICACIÓN

Los miembros del tribunal, luego de haber receptado la Defensa de trabajo escrito, hemos determinado la siguiente calificación.

Para constancia de lo expuesto firman:

Ing. Fernando Molina

-

Presidente

Ing. Danny Velasco

Director

Ing. Jorge Delgado

-

Miembro

Nota: (*SOBRE DIEZ*)

AUDITORIA DE LA INVESTIGACIÓN

Nosotros, Marco Aníbal Baldeón López. y Carlos Fabricio. Narváez Gavilanes. Somos responsables de las ideas, doctrinas y resultados expuestos en el presente trabajo de investigación, y los derechos de autoría que pertenecen a la Universidad Nacional de Chimborazo.

AGEADECIMIENTO

A la Facultad de Ingeniería de la Universidad Nacional de Chimborazo y de manera especial al Ing. Danny Velasco Director de la Tesis, al Ing. Fernando Molina Asesor de Tesis, al Ing. Jorge Delgado miembro del Tribunal, quienes amablemente nos supieron colaborar con sus conocimientos para culminar exitosamente nuestra tesis de grado. A nuestras familias por su amor paciencia entendimiento y continuo empuje.

Y a todas las personas con quienes trabajamos.

DEDICATORIA

A Dios por permitirme culminar la carrera, a mis Padres que aunque ya no estén presentes, sé que siempre están a mi lado, a mi Esposa María Elena y a mi hija Camila que están ahí siempre apoyándome con la fuerza de su presencia, a mi familia y de manera muy especial a mi querido hermano Wilson, pilar fundamental de éste logro.

Marco Aníbal Baldeón López

Dios, por darme la oportunidad de vivir y por estar conmigo en cada paso que doy, por fortalecer mi corazón e iluminar mi mente y por haber puesto en mi camino a aquellas personas que han sido mi soporte y compañía durante todo el periodo de estudio.

Mi madre, por darme la vida, quererme mucho, creer en mí y porque siempre me apoyaste. A mi querido hermano por estar siempre a mi lado.

Carlos Fabricio Narvárez Gavilanes

ÍNDICE

RESUMEN.....	16
SUMMARY	17
INTRODUCCIÓN	18
CAPITULO I.....	19
MARCO REFERENCIAL	19
1.1 TÍTULO DEL PROYECTO.....	20
1.2 PROBLEMATIZACIÓN	20
1.2.1 IDENTIFICACIÓN Y DESCRIPCIÓN DEL PROBLEMA	20
1.2.2 ANÁLISIS CRÍTICO	21
1.2.3 PROGNOSIS.....	21
1.2.4 DELIMITACIÓN.....	21
1.2.5 FORMULACIÓN DEL PROBLEMA.....	22
1.3 OBJETIVOS.....	22
1.3.1 GENERAL	22
1.3.2 ESPECÍFICOS	22
1.4 JUSTIFICACIÓN.....	23
CAPITULO II	24
FUNDAMENTACIÓN TEÓRICA.....	24
2.1 ANTECEDENTES DEL TEMA.....	25
2.1.1 AUTOEVALUACIÓN CURRICULAR EN EDUCACIÓN.....	25
2.2 ENFOQUE TEÓRICO	26
2.2.1 DEFINICIÓN DE CALIDAD DEL SOFTWARE	26
2.2.2 METODOLOGÍAS PARA DESARROLLO DE SOFTWARE.....	32
2.2.3 TIPOS DE METODOLOGÍAS.....	32
2.2.4 ENFOQUES DE DESARROLLO DE SOFTWARE	43
2.3 HERRAMIENTAS DE DESARROLLO DE SOFTWARE	47
2.4 HIPÓTESIS	47

2.5	IDENTIFICACIÓN DE VARIABLES	48
2.5.1	VARIABLE INDEPENDIENTE.....	48
2.5.2	VARIABLE DEPENDIENTE.....	48
CAPITULO III.....		49
PROCESO DE INGENIERIA DE SOFTWARE DEL SISTEMA.....		49
3.1	EVALUACIÓN CURRICULAR EN EDUCACIÓN.	50
3.2	ENFOQUE TEÓRICO	51
3.2.1	DEFINICIÓN DE CALIDAD DEL SOFTWARE	51
3.2.2	QUE SON LAS NORMAS ISO	51
3.2.3	LA FAMILIA ISO.....	52
3.2.4	COMO INFLUYEN UNA NORMA ISO EN EL DESARROLLO DE SOFTWARE.....	53
3.3	HERRAMIENTAS Y LENGUAJES EMPLEADOS EN EL DESARROLLO DEL SISTEMA SULTANA	58
CAPITULO IV.....		73
DESARROLLO DEL SISTEMA		73
4.1	IMPLANTACION DEL SISTEMA DE EVALUACIÓN CURRICULAR	74
4.1.1	DEFINICIÓN DEL PROBLEMA	74
4.1.2	DEFINIR ALTERNATIVAS DE SOLUCIÓN	74
4.1.3	DEFINICIÓN DEL ÁMBITO DEL SOFTWARE	75
4.2	DESCRIPCIÓN GENERAL	75
4.2.1	PERSPECTIVA DEL PRODUCTO.....	75
4.3	DESCRIPCIÓN DEL COLEGIO SULTANA DEL OREINTE.....	76
4.3.1	MISIÓN INSTITUCIONAL	76
4.3.2	VISIÓN INSTITUCIONAL	76
4.3.3	CREACIÓN	77
4.3.4	MODELO PEDAGÓGICO	78

4.3.5	GESTION ADMINISTRATIVA.....	78
4.3.6	GESTIÓN ACADÉMICA.....	79
4.4	SELECCIÓN Y JUSTIFICACION DE LA METODOLOGÍA.....	80
4.4.1	COMPARATIVA ENTRE METODOLOGÍAS DE DESARROLLO.....	82
4.4.2	CONCLUSIONES PARA SELECCIÓN DE METODOLOGÍA	82
4.4.3	SELECCIÓN Y JUSTIFICACIÓN DE LAS HERRAMIENTAS DE DESARROLLO	83
4.4.4	SERVIDOR DE APLICACIONES WEB.....	84
4.4.5	GESTOR DE BASE DE DATOS.....	84
4.4.6	LENGUAJE DE PROGRAMACIÓN	84
4.4.7	ENTORNO DE DESARROLLO INTEGRADO (IDE)	85
4.4.8	ESTÁNDARES Y BUENAS PRÁCTICAS PARA ENTORNOS WEB..	85
4.4.9	DISEÑO DEL CONTENIDO.....	86
4.5	ANÁLISIS Y DISEÑO DEL SISTEMA	91
4.5.1	ESPECIFICACIÓN DE REQUERIMIENTOS	91
4.5.2	USUARIOS DEL SISTEMA	91
4.6	ANÁLISIS DEL SISTEMA.....	101
4.6.1	ESTIMACIÓN DE ESFUERZOS	101
4.6.2	PRIORIZACIÓN.....	102
4.6.3	PLAN DE ENTREGAS	104
4.7	DISEÑO DEL SISTEMA.....	105
4.7.1	DISEÑO ARQUITECTÓNICO DEL SISTEMA	105
4.7.2	DISEÑO DEL DIAGRAMA DE CLASES	106
4.7.3	DISEÑO DE DIAGRAMAS DE INTERACCIÓN	108
4.7.4	GESTIÓN DE MATERIAS A EVALUAR.....	109
4.7.5	GESTIÓN DE REPORTE DE RESULTADOS	112
4.7.6	DISEÑO DEL ESQUEMA JERÁRQUICO	113

4.7.7	DISEÑO DE LAS INTERFACES DE USUARIO	114
4.8	IMPLANTACIÓN, PRUEBAS Y ANÁLISIS DE RESULTADOS ...	115
4.8.1	IMPLANTACIÓN DEL SISTEMA	115
4.9	PRUEBAS DEL SISTEMA	117
4.9.1	ELEMENTOS GRÁFICOS.....	118
CAPITULO V		126
METODOLOGIA		126
5.1	TIPO DE ESTUDIO.....	127
5.2	POBLACIÓN Y MUESTRA	127
5.3	OPERACIONALIZACIÓN DE VARIABLES.....	127
5.4	PROCEDIMIENTOS	128
5.4.1	TÉCNICA DE INVESTIGACIÓN.....	128
5.4.2	INSTRUMENTOS DE RECOLECCIÓN DE DATOS	128
5.5	PROCESAMIENTO Y ANÁLISIS	129
5.6	HIPÓTESIS	129
5.7	RESULTADOS	129
5.7.1	VARIABLE INDEPENDIENTE.....	129
5.8	ANÁLISIS COMPARATIVO DE LAS TÉCNICAS PARA EL CUMPLIMIENTO DE LA PANIFICACION ANUAL CURRICULAR.....	131
5.9	ANÁLISIS E INTERPRETACIÓN DE RESULTADOS.....	144
5.9.1	VARIABLE DEPENDIENTE.....	145
5.10	PRUEBA DE LA HIPOTESIS.....	160
5.11	DISCUSIÓN.....	162
CAPITULO VI.....		164
CONCLUSIONES Y RECOMENDACIONES.....		164
6.1	CONCLUSIONES.....	165
6.2	RECOMENDACIONES	167

GLOSARIO.....	168
BIBLIOGRAFIA.....	169
ANEXO A.....	170
ANEXO B.....	172
ANEXO C.....	177
MANUAL TÉCNICO.....	177
MANUAL DE USUARIO.....	187

ÍNDICE DE ILUSTRACIONES

Ilustración 1 Fases de un proyecto en eXtreme Programming	41
Ilustración 2 Parámetros de Calidad	54
Ilustración 3.-Diseño arquitectónico en tres capas.....	106
Ilustración 4 Diagrama de clases- Módulo de evaluación	107
Ilustración 5 Diagrama de Interacción-Gestión de Usuarios	108
Ilustración 6 Diagrama de Interacción-Ingreso de materias	109
Ilustración 7 Gestión de Evaluación-Ingreso de unidades	110
Ilustración 8 Gestión de Evaluación-Ingreso de temas	111
Ilustración 9 Gestión de Evaluación-Ingreso de Unidad a Evaluar	112
Ilustración 10 Reportes-Reporte Parcial de Resultados	112
Ilustración 11 Diagrama Jerárquico	113
Ilustración 12 Gestión de Usuarios	114
Ilustración 13 Gestión de Docentes.....	114
Ilustración 14 Gestión de Estudiantes	114
Ilustración 15 Gestión de Materias	115
Ilustración 16 Gestión de Evaluación Curricular.....	115
Ilustración 17 Prueba de compatibilidad con Microsoft Internet Explorer.....	120
Ilustración 18 Prueba de compatibilidad con Mozilla Firefox.....	121
Ilustración 19 Prueba de compatibilidad con Google Chrome	122
Ilustración 20 Prueba de compatibilidad con Opera	123
Ilustración 21 Gestión de Especialidades.....	170
Ilustración 22 Gestión de Cursos	170
Ilustración 23 Gestión de Paralelos.....	170
Ilustración 24 Asignación de Docentes.....	171
Ilustración 25 Gestión de Matrículas	171

Ilustración 26 Diagrama de Interacción-Asignación de Docentes.....	172
Ilustración 27 Diagrama de Interacción-Gestión de alumnos.....	173
Ilustración 28 Diagrama de Interacción-Ingreso de notas.....	174
Ilustración 29 Diagrama de Interacción-Docentes.....	175
Ilustración 30 Diagrama de Interacción-Matrículas.....	176
Ilustración 31 Gestión de la base de datos "Sultana"	179
Ilustración 32 CARPETA APPSERV.....	180
Ilustración 33 La carpeta www donde el apache busca la página index.....	180
Ilustración 34 Carpeta suloriente donde se ubicara el sitio.....	180
Ilustración 35 Carpeta style estilos css utilizados en el sistema.....	181
Ilustración 36 Carpeta de archivos php;.....	181
Ilustración 37 carpeta de imágenes e iconos utilizados	181
Ilustración 38 carpeta Galería donde se ha guardado las fotos de la galería.....	182
Ilustración 39 Notepad++.....	184
Ilustración 40 Subiendo el sitio y administrando por medio de servidor ftp	185

INDICE DE TABLAS

Tabla 1 Lenguajes soportados por Notepad++	69
Tabla 2 Metodologías ágiles vs. Metodologías tradicionales	81
Tabla 3 Comparación de metodologías.....	82
Tabla 4 Historia de Usuario -: Ingreso de estudiantes y profesores.....	94
Tabla 5 Historia de Usuario:-Creación de usuarios en el Sistema.....	94
Tabla 6 Historia de Usuario:-Ingreso de Cursos.....	95
Tabla 7 Historia de Usuario:-Matriculación.....	95
Tabla 8 Historia de Usuario:-Asignación de Docentes.....	96
Tabla 9 Historia de Usuario:-registro de Notas por Alumno.....	96
Tabla 10 Historia de usuario:-Ingreso de unidades y temas	96
Tabla 11 Historia de Usuario:-Ingreso de Materias.....	97
Tabla 12 Historia de Usuario:-Ingreso de Especialidades.....	97
Tabla 13 Historia de Usuario:-Ingreso de Paralelos	97
Tabla 14 Historia de Usuario:-Ingreso de Unidades y Temas	98
Tabla 15 Historia de Usuario:-Ingreso de Unidad a ser Evaluada.....	98
Tabla 16 Historia de Usuario:-Evaluación de Unidades por Alumno.....	99
Tabla 17 Historia de Usuario:-Reporte de Listado de Cursos.....	99
Tabla 18 Nómina General de Estudiantes.....	99
Tabla 19 Historia de Usuario:-Nómina General de Docentes.....	100
Tabla 20 Historia de Usuario:-Reporte de Resultados de Evaluación Curricular	100
Tabla 21 Historia de Usuario:-Envío de Tareas a los Alumnos.....	101
Tabla 22 Historia de Usuario:-Envío de Notificaciones a los alumnos	101
Tabla 23 Estimación de Esfuerzos	102
Tabla 24 Priorización	103

Tabla 25 Plan de Entregas.....	105
Tabla 26 Calendario de Entregas	105
Tabla 27 Especificación de hardware usado en Pruebas Unitarias	117
Tabla 28 Especificación de conexión de red utilizada en pruebas.....	117
Tabla 29 Especificación del Sistema Operativo en las Pruebas.....	118
Tabla 30 Formatos de imágenes utilizados	119
Tabla 31 Operacionalización de variables	128
Tabla 32 Correspondencias para Análisis de la Variable Independiente.....	145

GRÁFICOS

Gráfico 1 Resultados-funcionalidad.....	132
Gráfico 2 Resultados-Fiabilidad	134
Gráfico 3 Resultados-Usabilidad	136
Gráfico 4 Resultados-Eficiencia	138
Gráfico 5 Resultados-Mantenibilidad	140
Gráfico 6 Resultados-Portabilidad	142
Gráfico 7 Eficiencia	146
Gráfico 8 Eficacia	147
Gráfico 9 Efectividad	148
Gráfico 10 Capacidad de ser entendido	149
Gráfico 11 Capacidad de ser aprendido	150
Gráfico 12 Capacidad de ser operado	151
Gráfico 13 Capacidad de abstracción.....	152
Gráfico 14 Cumplimiento de usabilidad	153
Gráfico 15 Confiabilidad.....	154
Gráfico 16 Mantenibilidad	155
Gráfico 17 Calidad	156
Gráfico 18 Costo	157
Gráfico 19 Plazo.....	158
Gráfico 20 Factor Humana.....	159

RESUMEN

Hoy en día el Internet se ha vuelto el medio de vida, estilo y necesidad de las personas y empresas, el internet y la web son los avances más importantes de nuestra época. Ahora esta tecnología forma parte de nuestra vida diaria, debido a estas tecnologías como son la Norma ISO 9216 para el desarrollo y mejoramiento de Aplicaciones Web.

En este estudio se revisó la arquitectura, requerimientos, características, estructura organizativa, instalación, configuración y su respectiva administración.

El Sistema de Evaluación Curricular es un sistema multiplataforma basado en PHP que utiliza My SQL como gestor de base de datos y el servidor web Apache en Linux o en Windows, trabaja con plantillas las que se pueden crear desde cero o descargar desde el sitio Web, así mismo con los módulos que ofrece, facilitando la construcción del sitio y su adecuada ventaja de separación del diseño y contenido, presenta una interface amigable para el desarrollo de componentes y es la más completa para la administración.

Entre las metodologías para desarrollo de software se tomó en cuenta la Norma ISO 9216 cuyos parámetros de evaluación que se tomó en cuenta fueron: Requisitos del Sistema, Aplicaciones Disponibles, Soporte, Interoperabilidad, Comercio y Administración.

Empresas y organizaciones reconocidas internacionalmente ha migrado los contenidos de su web aplicando esta Norma ISO, lo cual indica que es suficiente por su estabilidad, funcionalidad y seguridad para llevar a cabo sistemas complejos.

SUMMARY

At present the Internet has become a way of living, style and needs of people and businesses or enterprises, the Internet and the Web are the most important developments of our time. Nowadays, this technology is a part of our daily lives because these technologies ISO 9216 System for Web application development was developed.

In this study we checked the architecture, requirements, characteristics, organizational structure, installation, configuration and administration.

This system is a PHP-based platform using My SQL as a data base manager and the Apache web server on Linux or Windows, it works with templates that can be created from the beginning or can be downloaded from the Drupal website, in the same way, with the modules offered, it facilitates the construction of the site and its adequate separation design and content, it has a friendly interface for the development of components and is the most complete for the administration.

We did a comparative study among existing in the market by using the tool which allowed us, the evaluation parameters used were: System Requirements, Available Applications, Support, Interoperability, Commerce and Management.

Enterprises and organizations recognized internationally like: ISO 9216l, which indicates that this is a system enough stability, functionality and security for carrying out complex systems.

INTRODUCCIÓN

En los últimos tiempos el Internet se ha vuelto el medio de vida, estilo y necesidad de las personas y empresas, siendo el avance más importante de nuestra época, esta tecnología forma parte de nuestra vida diaria, Por esta razón la Web necesita la aplicación de conceptos de ingeniería y sistemas de gestión de contenidos a fin de llevar al éxito el desarrollo de sistemas Web que estén al servicio de la sociedad y que no se pueden dar el lujo de fallar porque ocasionaría grandes pérdidas y por ende la desconfianza de la gente.

En los últimos años se ha desarrollado el concepto de sistema de gestión de contenidos, estos gestores de contenidos proporcionan un entorno que posibilita la actualización, mantenimiento y ampliación de la Web con la colaboración de múltiples usuarios.

En cualquier entorno virtual ésta es una característica importante, que además puede ayudar a crear una comunidad relacionada que participe más de forma conjunta, así como también es casi inimaginable tener como estrategia de desarrollo, la construcción de sitios Web desde cero, totalmente a medida, y sin ningún gestor de contenidos o plataforma de desarrollo.

En esta investigación se analizará el sistema de evaluación curricular teniendo en cuenta su arquitectura, los servicios prestados así como también sus fortalezas y debilidades, en base al análisis comparativo entre los sistemas anteriores y este.

Se aplicará la Norma ISO 9216, el cual se utiliza para mejoramiento de Aplicaciones Web y desarrollo de Software en general.

La Aplicación Web es una herramienta que facilitará el control del cumplimiento de objetivos, la explicación, la toma de decisiones y acciones de la Especialidad de Informática del Colegio Particular a Distancia Sultana del Oriente.

La evaluación se llevará a cabo por parte de los estudiantes del Tercer Año de Bachillerato Especialidad Informática. La presentación de reportes será analizada por la Junta Directiva y Docentes para tomar las medidas pertinentes a tiempo.

CAPITULO I
MARCO REFERENCIAL

1.1 TÍTULO DEL PROYECTO

Implantación del Sistema de Evaluación Curricular para el Bachillerato Informático del Colegio Sultana del Oriente de la ciudad de Macas en entorno WEB, aplicando la norma ISO/IEC 9216.

1.2 PROBLEMATIZACIÓN

1.2.1 IDENTIFICACIÓN Y DESCRIPCIÓN DEL PROBLEMA

Debido a las reformas educativas lanzadas por el actual gobierno y, en particular, la evaluación de un plantel educativo por medio del resultado de la evaluación de los alumnos de los terceros años del bachillerato mediante el examen de ingreso a las universidades del país, hace imprescindible un sistema automatizado que permita conocer y evaluar el cumplimiento riguroso del plan académico.

En la actualidad el Colegio Sultana del Oriente no posee un software que mejore la automatización de sus procesos académicos, lo que ocasiona gran pérdida de tiempo en la realización de actividades repetitivas y constantes en cada año lectivo.

No hay una base de datos con los profesores que son parte de la institución, también se planifica anualmente las materias a ser impartidas por los mismos a cada curso, detallando los horarios y los maestros involucrados, pero no existe un reporte de si se cumplió o no con los objetivos que permita mejorar la calidad de educación en la institución.

No existe estándar que mida la calidad de software desarrollo de software ágil que nos permita obtener una aplicación satisfactoria que cumpla con los requerimientos impuestos en el menor tiempo posible se vuelve una tarea complicada y en su mayoría no cumplen con ayudarnos a proporcionar aplicaciones de calidad.

1.2.2 ANÁLISIS CRÍTICO

La falta de un Sistema Informático que permita tener un seguimiento claro del nivel de cumplimiento de la malla curricular en los años de bachillerato que permita a los directivos tomar decisiones en la malla que se impartirá a los estudiantes durante el año lectivo tiene las siguientes causas:

- Desconocimiento de tecnologías modernas.
- Crecimiento de la demanda de estudiantes.
- Falta de coordinación, gestión académica.
- Falta de definición de los procesos de diseño curricular.

Las cuales concluyen en:

- Mala imagen institucional.
- Estudiantes y padres de familia insatisfechos.
- Pérdida de estudiantes
- Bajo nivel académico
- Procesos de control ineficientes

1.2.3 PROGNOSIS

Si no se implementa un sistema informático para la gestión y seguimiento curricular en el bachillerato en Informática podrían surgir los siguientes efectos: La mala imagen del colegio por no existir la inversión suficiente en tecnología por parte de directivos puede ocasionar la pérdida de estudiantes y más aún generar desconfianza a nivel institucional.

1.2.4 DELIMITACIÓN

El sistema se orienta exclusivamente al seguimiento y evaluación curricular del Bachillerato en Informática del tercer año de bachillerato informático del Colegio Sultana del Oriente de la ciudad de Macas.

Para tener un panorama claro del desarrollo del proyecto se hace necesario tomar muestras representativas por medio de encuestas a los estudiantes, autoridades y profesores para identificar los requerimientos de la aplicación.

Adicionalmente se trata de evaluar el cumplimiento de la malla curricular por parte de los alumnos y docentes del tercer año de bachillerato informático de la institución educativa.

1.2.5 FORMULACIÓN DEL PROBLEMA

¿Cómo incide la implementación del Sistema de Evaluación Curricular para el Bachillerato Informático del colegio Sultana del Oriente en el cumplimiento del plan anual curricular?

1.3 OBJETIVOS

1.3.1 GENERAL

Implantar el Sistema de Evaluación Curricular para el Bachillerato Informático del Colegio Sultana del Oriente de la ciudad de Macas en entorno WEB, aplicando la norma ISO/IEC 9216, para mejorar el cumplimiento del plan anual curricular.

1.3.2 ESPECÍFICOS

- Analizar el Sistema de Evaluación Curricular aplicado en el Colegio Sultana del oriente de la ciudad de macas.
- Generar software de Evaluación Curricular para el tercer año de bachillerato del colegio Sultana del Oriente que cumpla con normas de calidad a través de la aplicación de la norma ISO/IEC 9216.
- Analizar la norma ISO/IEC 9216 y su aplicabilidad en el desarrollo de Sistemas Curriculares.
- Llevar un control de la actividad docente.
- Evaluar la efectividad del sistema de Evaluación Curricular.

1.4 JUSTIFICACIÓN

La educación media es un punto fundamental en el desarrollo de las personas tanto intelectual como académicamente, por tanto se hace necesario mejorar la calidad de la educación a este nivel; además de que el país está atravesando una serie de cambios a nivel educacional creando una serie de requisitos para el ingreso de estudiantes a formar parte de las universidades de cualquier índole, uno de estos requerimientos es el examen de ingreso.

Entonces, ¿cómo mejorar el nivel académico de los educandos?. El problema principal es la falta de seguimiento de las actividades de los docentes con los educandos a través del ciclo escolar ya que en la mayoría de casos no se cumple con la malla curricular establecida.

De lo expuesto surge una nueva pregunta: Si tenemos tecnología ¿por qué no la utilizamos a nuestro bienestar?

Se debe aplicar la tecnología para crear herramientas ágiles que permitan dar seguimiento e identificar los problemas en el desarrollo del ciclo escolar y el cumplimiento de la malla curricular para de esta manera ir mejorando el desempeño de los alumnos, principalmente los que aspiren seguir una carrera universitaria.

El fácil acceso a herramientas de desarrollo de software libre orientado a la Web permite contar con los medios necesarios para desarrollar el proyecto de manera que no sea necesario el licenciamiento de Software reduciendo de esta manera el gasto de recursos económicos.

El desarrollo de software proporciona la capacidad de ofrecer al usuario del sistema una interface sencilla pero eficaz para el control y mediante la investigación profundizar los conocimientos en programación.

Se proporciona una herramienta para brindar seguridad, comodidad y rapidez en el control, prestando un servicio a favor del mejoramiento de la institución y sin precedentes en Macas.

CAPITULO II
FUNDAMENTACIÓN TEÓRICA

2.1 ANTECEDENTES DEL TEMA

Los proyectos de desarrollo de software de la actualidad se desenvuelven en un entorno muy cambiante donde se exige reducir los tiempos de desarrollo, manteniendo una alta calidad en su resultado final sin involucrar un aumento considerable en la complejidad de las fases en las que el proyecto es realizado.

La elección de una metodología adecuada que se ajuste al equipo de desarrollo permitirá una flexibilidad para la obtención de los objetivos definidos al inicio del proyecto y además proveerá satisfacción al cliente más allá de sus necesidades o expectativas, ya que las metodologías ágiles están orientadas para proyectos pequeños que requieran un tiempo corto para desarrollo, se las consideran como una solución a la medida que no abandona las prácticas tradicionales que aseguran la calidad del producto final que se entrega al usuario.

Es por esto que el enfoque que se dará al presente proyecto se basa en el desarrollo ágil de aplicaciones.

2.1.1 *AUTOEVALUACIÓN CURRICULAR EN EDUCACIÓN.*

Se presenta una propuesta metodológica para la evaluación del currículo en modelos a distancia, la cual será validada en la carrera de Informática del Colegio Particular a Distancia Sultana del Oriente en un proceso de autoevaluación.

Esta experiencia establece la necesidad de que los programas académicos inicien su proceso de mejoramiento continuo con el establecimiento de una línea base, realizando una evaluación en la que se diagnostica el estado de los planes de estudio y de cada uno de los cursos que lo integran. Posterior a esta evaluación la carrera se deberá iniciar un rediseño que dará como resultado un planteamiento curricular renovado que en adelante se mantendrá en cambio constante, sujeto al monitoreo a lo largo de la vida de la carrera.

Ante esta situación resulta necesario contar con una metodología y con instrumental para hacer una evaluación integral, un análisis riguroso de los

aspectos cruciales del plan de estudios y una clara identificación de las verdaderas debilidades y fortalezas sobre las cuales desarrollar el proceso de mejoramiento continuo.

Como respuesta a esta necesidad se diseñará una metodología que permitirá a los responsables desarrollar un proceso participativo de evaluación del currículo y al mismo tiempo un monitoreo constante para evaluar el cumplimiento de metas y de los contenidos que se encuentran en cada malla.

El proceso de evaluación curricular se desarrollará mediante una metodología participativa que involucrará al personal académico, al estudiando, graduados, empleadores y expertos en diferentes momentos tanto de la revisión como juzgamiento de la calidad de los aspectos curriculares de las carrera de informática.

2.2 ENFOQUE TEÓRICO

2.2.1 DEFINICIÓN DE CALIDAD DEL SOFTWARE

La Calidad del software es el desarrollo de software basado en estándares con la funcionalidad y rendimiento total que satisfacen los requerimientos del cliente.

Además dice que es un proceso de desarrollo, gestión de proyectos, análisis y diseño, especificación de requerimientos, arquitectura, son solo algunos de los componentes que se aglomeran para conformar la ingeniería de software (IS) como disciplina para la creación y mantenimiento de software. Dentro de ésta, existe un subconjunto de teorías, herramientas y métodos orientados a lo que se denomina la calidad del software. Para resumir de alguna manera la amplitud de este concepto, se puede decir que la calidad de software ha sido usada desde un simple argumento de venta, hasta verdaderos estudios formales y usos de métricas para el desarrollo de software. Extrañamente dentro de la IS, la calidad del software es muy complicada de definir y de enmarcar en un simple concepto teórico, por lo que en esta nota, me concentraré solo en las diversas características

que permiten describirla y en los elementos que importan específicamente al diseñador de software.

Una idea general sobre un software de calidad es aquel que debiera cumplir con los requerimientos funcionales y de performance además de ser mantenible, confiable y aceptable.

2.2.1.1 *Que son las normas ISO*

ISO es una federación mundial que agrupa a representantes de cada uno de los organismos nacionales de estandarización (como lo es el IRAM en la Argentina), y que tiene como objeto desarrollar estándares internacionales que faciliten el comercio internacional.

Cuando las organizaciones tienen una forma objetiva de evaluar la calidad de los procesos de un proveedor, el riesgo de hacer negocios con dicho proveedor se reduce en gran medida, y si los estándares de calidad son los mismos para todo el mundo, el comercio entre empresas de diferentes países puede potenciarse en forma significativa – y de hecho, así ha ocurrido –.

Durante las últimas décadas, organizaciones de todos los lugares del mundo se han estado preocupando cada vez más en satisfacer eficazmente las necesidades de sus clientes, pero las empresas no contaban, en general, con literatura sobre calidad que les indicara de qué forma, exactamente, podían alcanzar y mantener la calidad de sus productos y servicios.

De forma paralela, las tendencias crecientes del comercio entre naciones reforzaba la necesidad de contar con estándares universales de la calidad. Sin embargo, no existía una referencia estandarizada para que las organizaciones de todo el mundo pudieran demostrar sus prácticas de calidad o mejorar sus procesos de fabricación o de servicio.

Teniendo como base diferentes antecedentes sobre normas de estandarización que se fueron desarrollando principalmente en Gran Bretaña, la ISO creó y publicó en

1987 sus primeros estándares de dirección de la calidad: los estándares de calidad de la serie ISO 9000.

Con base en Ginebra, Suiza, esta organización ha sido desde entonces la encargada de desarrollar y publicar estándares voluntarios de calidad, facilitando así la coordinación y unificación de normas internacionales e incorporando la idea de que las prácticas pueden estandarizarse tanto para beneficiar a los productores como a los compradores de bienes y servicios. Particularmente, los estándares ISO 9000 han jugado y juegan un importante papel al promover un único estándar de calidad a nivel mundial.

2.2.1.2 La Familia Iso

Las series de normas ISO relacionadas con la calidad constituyen lo que se denomina familia de normas, las que abarcan distintos aspectos relacionados con la calidad:

ISO 9000: Sistemas de Gestión de Calidad Fundamentos, vocabulario, requisitos, elementos del sistema de calidad, calidad en diseño, fabricación, inspección, instalación, venta, servicio post venta, directrices para la mejora del desempeño.

ISO 10000: Guías para implementar Sistemas de Gestión de Calidad/ Reportes Técnicos Guía para planes de calidad, para la gestión de proyectos, para la documentación de los SGC, para la gestión de efectos económicos de la calidad, para aplicación de técnicas estadísticas en las Normas ISO 9000. Requisitos de aseguramiento de la calidad para equipamiento de medición, aseguramiento de la medición.

ISO 14000: Sistemas de Gestión Ambiental de las Organizaciones. Principios ambientales, etiquetado ambiental, ciclo de vida del producto, programas de revisión ambiental, auditorías.

ISO 19011: Directrices para la Auditoría de los SGC y/o Ambiental

2.2.1.3 *Como influyen una norma ISO en el desarrollo de software.*

ISO 9216 es un estándar internacional para la evaluación de la calidad del software. Está remplazado por el proyecto SQuaRE, ISO 25000:2005, el cual sigue los mismos conceptos.

El estándar está dividido en cuatro partes las cuales dirigen, respectivamente, lo siguiente: modelo de calidad, métricas externas, métricas internas y calidad en las métricas de uso.

El modelo de calidad establecido en la primera parte del estándar, ISO 9126-1, clasifica la calidad del software en un conjunto estructurado de características y subcaracterísticas de la siguiente manera:

- **Funcionalidad** - Un conjunto de atributos que se relacionan con la existencia de un conjunto de funciones y sus propiedades específicas. Las funciones son aquellas que satisfacen las necesidades implícitas o explícitas.
 - Idoneidad
 - Exactitud
 - Interoperabilidad
 - Seguridad
 - Cumplimiento de normas.
- **Fiabilidad** - Un conjunto de atributos relacionados con la capacidad del software de mantener su nivel de prestación bajo condiciones establecidas durante un período establecido.
 - Madurez
 - Recuperabilidad
 - Tolerancia a fallos
- **Usabilidad** - Un conjunto de atributos relacionados con el esfuerzo necesario para su uso, y en la valoración individual de tal uso, por un establecido o implicado conjunto de usuarios.

- Aprendizaje
 - Comprensión
 - Operatividad
 - Atractividad
- **Eficiencia** - Conjunto de atributos relacionados con la relación entre el nivel de desempeño del software y la cantidad de recursos necesitados bajo condiciones establecidas.
 - Comportamiento en el tiempo
 - Comportamiento de recursos
 - **Mantenibilidad** - Conjunto de atributos relacionados con la facilidad de extender, modificar o corregir errores en un sistema software.
 - Estabilidad
 - Facilidad de análisis
 - Facilidad de cambio
 - Facilidad de pruebas
 - **Portabilidad** - Conjunto de atributos relacionados con la capacidad de un sistema software para ser transferido desde una plataforma a otra.
 - Capacidad de instalación
 - Capacidad de reemplazamiento
 - Adaptabilidad
 - Co-Existencia

La subcaracterística Conformidad no está listada arriba ya que se aplica a todas las características. Ejemplos son conformidad a la legislación referente a usabilidad y fiabilidad.

Cada subcaracterística (como adaptabilidad) está dividida en atributos. Un atributo es una entidad la cual puede ser verificada o medida en el producto software. Los atributos no están definidos en el estándar, ya que varían entre diferentes productos software.

Un producto software está definido en un sentido amplio como: los ejecutables, código fuente, descripciones de arquitectura, y así. Como resultado, la noción de

usuario se amplía tanto a operadores como a programadores, los cuales son usuarios de componentes como son bibliotecas software.

El estándar provee un entorno para que las organizaciones definan un modelo de calidad para el producto software. Haciendo esto así, sin embargo, se lleva a cada organización la tarea de especificar precisamente su propio modelo. Esto podría ser hecho, por ejemplo, especificando los objetivos para las métricas de calidad las cuales evalúan el grado de presencia de los atributos de calidad.

Métricas internas son aquellas que no dependen de la ejecución del software (medidas estáticas).

Métricas externas son aquellas aplicables al software en ejecución.

La calidad en las métricas de uso están sólo disponibles cuando el producto final es usado en condiciones reales.

Idealmente, la calidad interna no necesariamente implica calidad externa y esta a su vez la calidad en el uso.

Este estándar proviene desde el modelo establecido en 1977 por McCall y sus colegas, los cuales propusieron un modelo para especificar la calidad del software. El modelo de calidad McCall está organizado sobre tres tipos de Características de Calidad:

- Factores (especificar): Describen la visión externa del software, como es visto por los usuarios.
- Criterios (construir): Describen la visión interna del software, como es visto por el desarrollador.
- Métricas (controlar): Se definen y se usan para proveer una escala y método para la medida.

ISO 9126 distingue entre fallo y no conformidad. Un fallo es el incumplimiento de los requisitos previos, mientras que la no conformidad es el incumplimiento de los requisitos especificados. Una distinción similar es la que se establece entre validación y verificación.

2.2.2 METODOLOGÍAS PARA DESARROLLO DE SOFTWARE.

Una metodología de desarrollo de software se refiere a un framework que es usado para estructurar, planear y controlar el proceso de desarrollo en sistemas de información.

A lo largo del tiempo, una gran cantidad de métodos han sido desarrollados diferenciándose por su fortaleza y debilidad.

El framework para metodología de desarrollo de software consiste en:

- Una filosofía de desarrollo de programas de computación con el enfoque del proceso de desarrollo de software
- Herramientas, modelos y métodos para asistir al proceso de desarrollo de software

Estos frameworks son a menudo vinculados a algún tipo de organización, que además desarrolla, apoya el uso y promueve la metodología. La metodología es a menudo documentada en algún tipo de documentación formal.

2.2.3 TIPOS DE METODOLOGÍAS.

Desde los años 70's hasta nuestros días se han ido desarrollando diferentes metodologías para desarrollar aplicaciones informáticas; a continuación se enlistan la mayoría de estas metodologías.

- Programación estructurada sol desde 1969
- Programación estructurada Jackson desde 1975
- Structured Systems Analysis and Design Methodology (SSADM) desde 1980
- Structured Analysis and Design Technique (SADT) desde 1980
- Ingeniería de la información (IE/IEM) desde 1981
- Rapid application development (RAD) desde 1991.
- Programación orientada a objetos (OOP) a lo largo de la década de los 90's
- Virtual finite state machine (VFSM) desde 1990s
- Dynamic Systems Development Method desarrollado en UK desde 1995.
- Scrum (desarrollo), en la última parte de los 90's
- Rational Unified Process (RUP) desde 1999.

- Extreme Programming(XP) desde 1999
- Enterprise Unified Process (EUP) extensiones RUP desde 2002
- Constructionist design methodology (CDM) desde 2004 por Kristinn R. Thórisson
- Agile Unified Process (AUP) desde 2005 por Scott Ambler

2.2.3.1 *eXtreme Programming.*

Esta metodología promueve los siguientes valores:

Comunicación.- El eXtreme Programming se nutre del ancho de banda más grande que se puede obtener cuando existe algún tipo de comunicación: la comunicación directa entre personas. Es muy importante entender cuáles son las ventajas de este medio. Cuando dos (o más) personas se comunican directamente pueden no solo consumir las palabras formuladas por la otra persona, sino que también aprecian los gestos, miradas, etc. que hace su compañero. Sin embargo, en una conversación mediante el correo electrónico, hay muchos factores que hacen de esta una comunicación, por así decirlo, mucho menos efectiva.

Coraje.- El coraje es un valor muy importante dentro de la programación extrema. Un miembro de un equipo de desarrollo extremo debe de tener el coraje de exponer sus dudas, miedos, experiencias sin "embellecer" éstas de ninguna de las maneras. Esto es muy importante ya que un equipo de desarrollo extremo se basa en la confianza para con sus miembros. Faltar a esta confianza es una falta más que grave.

Simplicidad.- Dado que no se puede predecir cómo va a ser en el futuro, el software que se está desarrollando; un equipo de programación extrema intenta mantener el software lo más sencillo posible. Esto quiere decir que no se va a invertir ningún esfuerzo en hacer un desarrollo que en un futuro pueda llegar a tener valor. En el XP frases como "...en un futuro vamos a necesitar..." o "Haz un sistema genérico de..." no tienen ningún sentido ya que no aportan ningún valor en el momento.

Feedback.- La agilidad se define (entre otras cosas) por la capacidad de respuesta ante los cambios que se van haciendo necesarios a lo largo del camino. Por este

motivo uno de los valores que agilizan el proceso es el continuo seguimiento o feedback que se recibe en un entorno ágil de desarrollo. Este feedback se toma del cliente, de los miembros del equipo, en cuestión de todo el entorno en el que se mueve un equipo de desarrollo ágil.

Para ello se fundamenta en las siguientes prácticas:

Planificación incremental.- La Programación Extrema asume que la planificación nunca será perfecta, y que variará en función de cómo varíen las necesidades del negocio. Por tanto, el valor real reside en obtener rápidamente un plan inicial, y contar con mecanismos de feedback que permitan conocer con precisión dónde estamos. Como es lógico, la planificación es iterativa: un representante del negocio decide al comienzo de cada iteración qué características concretas se van a implementar.

El objetivo de la XP es generar versiones de la aplicación tan pequeñas como sea posible, pero que proporcionen un valor adicional claro, desde el punto de vista del negocio. A estas versiones se las denomina releases.

Una release cuenta con un cierto número de historias. La historia es la unidad de funcionalidad en un proyecto XP, y corresponde a la mínima funcionalidad posible que tiene valor desde el punto de vista del negocio. Durante cada iteración se cierran varias historias, lo que hace que toda iteración añada un valor tangible para el cliente.

Es fundamental en toda esta planificación la presencia de un representante del cliente, que forma parte del equipo y que decide cuáles son las historias más valiosas. Estas historias son las que se desarrollarán en la iteración actual.

La obtención de feedback que permita llevar a cabo estimaciones precisas es fundamental. Se hacen estimaciones para cada historia, de modo que en cuanto se comienzan a tener datos históricos, éstos se utilizan para hacer que las siguientes estimaciones sean más precisas.

Como se puede ver, y como siempre ocurre con la Programación Extrema, el enfoque utilizado para llevar a cabo la planificación es eminentemente

pragmático. Gran parte de la eficacia de este modelo de planificación deriva de una división clara de responsabilidades, que tiene en cuenta las necesidades del negocio en todo momento. Dentro de esta división, el representante del cliente tiene las siguientes responsabilidades:

- Decidir qué se implementa en cada release o iteración.
- Fijar las fechas de fin de la release, recortando unas características o añadiendo otras.
- Priorizar el orden de implementación, en función del valor de negocio.

Las responsabilidades del equipo de desarrollo son las siguientes:

- Estimar cuánto tiempo llevará una historia: este feedback es fundamental para el cliente, y puede llevarle a reconsiderar qué historias se deben incluir en una iteración.
- Proporcionar información sobre el coste de utilizar distintas opciones tecnológicas.
- Organizar el equipo.
- Estimar el riesgo de cada historia.
- Decidir el orden de desarrollo de historias dentro de la iteración.

Testing.- La ejecución automatizada de tests es un elemento clave de la XP. Existen tanto tests internos (o tests de unidad), para garantizar que el mismo es correcto, como tests de aceptación, para garantizar que el código hace lo que debe hacer. El cliente es el responsable de definir los tests de aceptación, no necesariamente de implementarlos. Él es la persona mejor cualificada para decidir cuál es la funcionalidad más valiosa.

El hecho de que los tests sean automatizados es el único modo de garantizar que todo funciona: desde el punto de vista de la XP, si no hay tests, las cosas sólo funcionan en apariencia. Aún más, si un test no está automatizado, no se le puede considerar como tal.

El objetivo de los tests no es corregir errores, sino prevenirlos. Por ejemplo, los tests siempre se escriben antes que el código a testear, no después: esto aporta un

gran valor adicional, pues fuerza a los desarrolladores a pensar cómo se va a usar el código que escriben, poniéndolos en la posición de consumidores del software. Elaborar los tests exige pensar por adelantado cuáles son los problemas más graves que se pueden presentar, y cuáles son los puntos dudosos. Esto evita muchos problemas y dudas, en lugar de dejar que aparezcan "sobre la marcha".

Un efecto lateral importante de los tests es que dan una gran seguridad a los desarrolladores: es posible llegar a hacer cambios más o menos importantes sin miedo a problemas inesperados, dado que proporcionan una red de seguridad. La existencia de tests hace el código muy maleable.

Programación en parejas.- La XP incluye, como una de sus prácticas estándar, la programación en parejas. Nadie programa en solitario, siempre hay dos personas delante del ordenador. Ésta es una de las características que más se cuestiona al comienzo de la adopción de la XP dentro de un equipo, pero en la práctica se acepta rápidamente y de forma entusiasta.

El principal argumento contra la programación en parejas es que es improductiva. Esta idea se basa en el hecho de que dos programadores "programan el doble por separado". Esto sería así si no fuera por las siguientes razones:

- El hecho de que todas las decisiones las tomen al menos dos personas proporciona un mecanismo de seguridad enormemente valioso.
- Con dos personas responsabilizándose del código en cada momento, es menos probable que se caiga en la tentación de dejar de escribir tests, etc., algo fundamental para mantener el código en buena forma. Es muy difícil que dos personas se salten tareas por descuido o negligencia.
- El hecho de programar en parejas permite la dispersión de know-how (conocimiento fundamental) por todo el equipo. Este efecto es difícil de conseguir de otro modo, y hace que la incorporación de nuevos miembros al equipo sea mucho más rápida y eficaz.
- El código siempre está siendo revisado por otra persona. La revisión de código es el método más eficaz de conseguir código de calidad, algo corroborado por numerosos estudios, muchos de los cuáles son anteriores a la Programación Extrema.

- En contra de lo que pueda parecer, los dos desarrolladores no hacen lo mismo: mientras el que tiene el teclado adopta un rol más táctico, el otro adopta un rol más estratégico, preguntándose constantemente si lo que se está haciendo tiene sentido desde un punto de vista global.
- Los datos indican que la programación en parejas es realmente más eficiente. Si bien se sacrifica un poco de velocidad al comienzo, luego se obtiene una velocidad de crucero muy superior. Esto contrasta con lo que ocurre en la mayor parte de los proyectos, en los que se arranca con una velocidad enorme pero rápidamente se llega a un estado muy parecido a la parálisis, en el que progresos cada vez más pequeños consumen cantidades de tiempo cada vez más grandes. Todos conocemos proyectos que se pasan el 50% del tiempo en el estado de "finalizado al 90%".

Refactorización.- A la hora de la verdad, el código de la mayor parte de las aplicaciones empieza en un razonable buen estado, para luego deteriorarse de forma progresiva. El coste desorbitado del mantenimiento, modificación y ampliación de aplicaciones ya existente se debe en gran parte a este hecho.

Uno de los objetivos de la XP es mantener la curva de costes tan plana como sea posible, por lo que existen una serie de mecanismos destinados a mantener el código en buen estado, modificándolo activamente para que conserve claridad y sencillez. A este proceso básico para mantener el código en buena forma se le llama refactorización.

La refactorización no sólo sirve para mantener el código legible y sencillo: también se utiliza cuando resulta conveniente modificar código existente para hacer más fácil implementar nueva funcionalidad.

Diseño simple.- Otra práctica fundamental de la Programación Extrema es utilizar diseños tan simples como sea posible. El principio es "utilizar el diseño más sencillo que consiga que todo funcione". Se evita diseñar características extra porque a la hora de la verdad la experiencia indica que raramente se puede anticipar qué necesidades se convertirán en reales y cuáles no. La XP nos pide que no vivamos bajo la ilusión de que un diseño puede resolver todas o gran parte de las situaciones futuras: lo que parece necesario cambia con frecuencia, es difícil acertar a priori.

Es obvio que, si no vamos a anticipar futuras necesidades, debemos poder modificar el diseño si alguna de estas se materializa. La XP soporta estas modificaciones gracias a los tests automatizados. Estos permiten hacer cambios importantes gracias a la red de protección que proporcionan. La refactorización, que hace que el código existente sea claro y sencillo, también ayuda a hacer factibles las modificaciones.

La XP define un "diseño tan simple como sea posible" como aquél que:

- Pasa todos los tests.
- No contiene código duplicado.
- Deja clara la intención de los programadores (enfatisa el qué, no el cómo) en cada línea de código.
- Contiene el menor número posible de clases y métodos.

Propiedad colectiva del código.- La XP aboga por la propiedad colectiva del código. En otras palabras, todo el mundo tiene autoridad para hacer cambios a cualquier código, y es responsable de ellos. Esto permite no tener que estar esperando a otros cuando todo lo que hace falta es algún pequeño cambio.

Por supuesto, cada cuál es responsable de las modificaciones que haga. El principio básico es "tú lo rompes, tú lo arreglas, no importa si está en el código propio o en el de otros".

Por último, vale la pena tener en cuenta que la existencia de tests automatizados impide que se produzca un desarrollo anárquico, al ser cada persona responsable de que todos los tests se ejecuten con éxito al incorporar los cambios que ha introducido al programa.

Integración continua.- En muchos casos la integración de código produce efectos laterales imprevistos, y en ocasiones la integración puede llegar a ser realmente traumática, cuando dejan de funcionar cosas por motivos desconocidos. La Programación Extrema hace que la integración sea permanente, con lo que todos los problemas se manifiestan de forma inmediata, en lugar de durante una fase de integración más o menos remota.

La existencia de una fase de integración separada tiene dos efectos laterales indeseables: se empieza a hacer codificación "yo-yo", en la que todo el mundo modifica código "sólo para que funcione, ya lo ajustaremos", y hace que se acumulen defectos. Evitar que se acumulen defectos es muy importante para la XP, como lo es el conseguir que los defectos que cada programador inyecta los elimine él mismo.

Cliente en el equipo.- Algunos de los problemas más graves en el desarrollo son los que se originan cuando el equipo de desarrollo toma decisiones de negocio críticas. Esto no debería ocurrir, pero a la hora de la verdad con frecuencia no se obtiene feedback del cliente con la fluidez necesaria: el resultado es que se ha de optar por detener el avance de los proyectos. Por otra parte, los representantes del negocio también suelen encontrarse con problemas inesperados debido a que tampoco reciben el feedback adecuado por parte de los desarrolladores.

La XP intenta resolver este tipo de problemas integrando un representante del negocio dentro del equipo de desarrollo. Ésta persona siempre está disponible para resolver dudas y para decidir qué y qué no se hace en cada momento, en función de los intereses del negocio. Debido a su inmersión dentro del equipo, y a que es él quien decide qué y qué no se hace, junto con los tests que verifican si la funcionalidad es la correcta y deseada, esta persona obtiene un feedback absolutamente realista del estado del proyecto.

Releases pequeñas.- Siguiendo la política de la XP de dar el máximo valor posible en cada momento, se intenta liberar nuevas versiones de las aplicaciones con frecuencia. Éstas deben ser tan pequeñas como sea posible, aunque deben añadir suficiente valor como para que resulten valiosas para el cliente.

Semanas de 40 horas.- La Programación Extrema lleva a un modo de trabajo en que el equipo está siempre al 100%. Una semana de 40 horas en las que se dedica la mayor parte del tiempo a tareas que suponen un avance puede dar mucho de sí, y hace innecesario recurrir a sobreesfuerzos -excepto en casos extremos.

Además, el sobreesfuerzo continuado pronto lleva a un rendimiento menor y a un deterioro de la moral de todo el equipo.

Estándares de codificación.- Para conseguir que el código se encuentre en buen estado y que cualquier persona del equipo pueda modificar cualquier parte del código es imprescindible que el estilo de codificación sea consistente. Un estándar de codificación es necesario para soportar otras prácticas de la XP.

Sin embargo, la XP también es pragmática en esto, y apuesta por establecer un número mínimo de reglas: el resto se irán pactando de-facto. Esto evita un ejercicio inicial más o menos estéril.

Uso de Metáforas.- La comunicación fluida es uno de los valores más importantes de la Programación Extrema: la programación en parejas, el hecho de incorporar al equipo una persona que represente los intereses del negocio y otras prácticas son valiosas entre otras cosas porque potencian enormemente la comunicación.

Para conseguir que la comunicación sea fluida es imprescindible, entre otras cosas, utilizar el vocabulario del negocio. También es fundamental huir de definiciones abstractas. Dicho de otro modo, la XP no pretende seguir la letra de la ley, sino su espíritu. Dentro de este enfoque es fundamental buscar continuamente metáforas que comuniquen intenciones y resulten descriptivas, enfatizando el "qué" por delante del "cómo".

La metodología XP es una metodología *ágil*.

- Los individuos e interacciones son más importantes que los procesos y herramientas.

Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.

- Software que funcione es más importante que documentación exhaustiva.

Desarrollar software que funciona más que conseguir una buena documentación. La regla a seguir es “no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante”. Estos documentos deben ser cortos y centrarse en lo fundamental.

- La colaboración con el cliente es más importante que la negociación de contratos.

La colaboración con el cliente más que la negociación de un contrato. Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.

- La respuesta ante el cambio es más importante que el seguimiento de un plan.

Responder a los cambios más que seguir estrictamente un plan. La habilidad de responder a los cambios que puedan surgir a los largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

2.2.3.1.1 Ciclo de vida de un proyecto XP

Las fases que define *eXtreme Programming* pueden verse en la siguiente figura,



Ilustración 1 Fases de un proyecto en eXtreme Programming

El ciclo de vida ideal de XP consiste de seis fases.

1. Exploración

En esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo. La fase de exploración toma de pocas semanas a pocos meses, dependiendo del tamaño y familiaridad que tengan los programadores con la tecnología.

2. Planificación de la Entrega (*Release*)

En esta fase el cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días. Las estimaciones de esfuerzo asociado a la implementación de las historias la establecen los programadores utilizando como medida el punto. Un punto, equivale a una semana ideal de programación. Las historias generalmente valen de 1 a 3 puntos. Por otra parte, el equipo de desarrollo mantiene un registro de la “velocidad” de desarrollo, establecida en puntos por iteración, basándose principalmente en la suma de puntos correspondientes a las historias de usuario que fueron terminadas en la última iteración. La planificación se puede realizar basándose en el tiempo o el alcance. La velocidad del proyecto es utilizada para establecer cuántas historias se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementar un conjunto de historias. Al planificar por tiempo, se multiplica el número de iteraciones por la velocidad del proyecto, determinándose cuántos puntos se pueden completar. Al planificar según alcance del sistema, se divide la suma de puntos de las historias de usuario seleccionadas entre la velocidad del proyecto, obteniendo el número de iteraciones necesarias para su implementación.

3. Iteraciones

Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado. El Plan de Entrega está compuesto por iteraciones de no más de tres semanas. En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada durante el resto del proyecto. Esto se logra escogiendo las historias que fueren la creación de esta arquitectura, sin embargo, esto no siempre es posible ya que es el cliente quien decide qué historias se implementarán en cada iteración (para maximizar el valor de negocio). Al

final de la última iteración el sistema estará listo para entrar en producción. Los elementos que deben tomarse en cuenta durante la elaboración del Plan de la Iteración son: historias de usuario no abordadas, velocidad del proyecto, pruebas de aceptación no superadas en la iteración anterior y tareas no terminadas en la iteración anterior. Todo el trabajo de la iteración es expresado en tareas de programación, cada una de ellas es asignada a un programador como responsable, pero llevadas a cabo por parejas de programadores.

4. Producción

La fase de producción requiere de pruebas adicionales y revisiones de rendimiento antes de que el sistema sea trasladado al entorno del cliente. Al mismo tiempo, se deben tomar decisiones sobre la inclusión de nuevas características a la versión actual, debido a cambios durante esta fase. Es posible que se rebaje el tiempo que toma cada iteración, de tres a una semana. Las ideas que han sido propuestas y las sugerencias son documentadas para su posterior implementación (por ejemplo, durante la fase de mantenimiento).

5. Mantenimiento

Mientras la primera versión se encuentra en producción, el proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones. Para realizar esto se requiere de tareas de soporte para el cliente. De esta forma, la velocidad de desarrollo puede bajar después de la puesta del sistema en producción. La fase de mantenimiento puede requerir nuevo personal dentro del equipo y cambios en su estructura.

6. Muerte del Proyecto

Es cuando el cliente no tiene más historias para ser incluidas en el sistema. Esto requiere que se satisfagan las necesidades del cliente en otros aspectos como rendimiento y confiabilidad del sistema. Se genera la documentación final del sistema y no se realizan más cambios en la arquitectura. La muerte del proyecto también ocurre cuando el sistema no genera los beneficios esperados por el cliente o cuando no hay presupuesto para mantenerlo.

2.2.4 ENFOQUES DE DESARROLLO DE SOFTWARE

Debido a que cada metodología tiene su propio enfoque de desarrollo, se podría definir algunos enfoques generales los cuales se nombran a continuación:

- Modelo en cascada: Framework lineal.
- Prototipado: Framework iterativo.

- Incremental: Combinación de framework lineal e iterativo.
- Espiral: Combinación de framework lineal e iterativo.
- RAD: Rapid Application Development, framework iterativo.

2.2.4.1 *Modelo en cascada.*

Es un proceso secuencial de desarrollo en el que los pasos de desarrollo son vistos hacia abajo (como en una cascada de agua) a través de las fases de análisis de las necesidades, el diseño, implementación, pruebas (validación), la integración, y mantenimiento.

Los principios básicos del modelo de cascada son los siguientes:¹

- El proyecto está dividido en fases secuenciales, con cierta superposición y splashback aceptable entre fases.
- Se hace hincapié en la planificación, los horarios, fechas, presupuestos y ejecución de todo un sistema de una sola vez.
- Un estricto control se mantiene durante la vida del proyecto a través de la utilización de una amplia documentación escrita, así como a través de comentarios y aprobación / signoff por el usuario y la tecnología de la información de gestión al final de la mayoría de las fases antes de comenzar la próxima fase.

2.2.4.2 *Prototipado*

El prototipado es el framework de actividades dedicada al desarrollo de software prototipo, es decir, versiones incompletas del software a desarrollar.

2.2.4.3 *Incremental*

Provee una estrategia para controlar la complejidad y los riesgos, desarrollando una parte del producto software reservando el resto de aspectos para el futuro.

Los principios básicos son:

- Una serie de mini-Cascadas se llevan a cabo, donde todas las fases de la cascada modelo de desarrollo se han completado para una pequeña parte de los sistemas, antes de proceder a la próxima incremental
- Se definen los requisitos antes de proceder con lo evolutivo, se realiza un mini-Cascada de desarrollo de cada uno de los incrementos del sistema
- El concepto inicial de software, análisis de las necesidades, y el diseño de la arquitectura y colectiva básicas se definen utilizando el enfoque de cascada, seguida por iterativo de prototipos, que culmina en la instalación del prototipo final.

2.2.4.4 *Espiral*

Los principios básicos son:

- La atención se centra en la evaluación y reducción del riesgo del proyecto dividiendo el proyecto en segmentos más pequeños y proporcionar más facilidad de cambio durante el proceso de desarrollo, así como ofrecer la oportunidad de evaluar los riesgos y con un peso de la consideración de la continuación del proyecto durante todo el ciclo de vida.
- Cada viaje alrededor de la espiral atraviesa cuatro cuadrantes básicos: (1) determinar objetivos, alternativas, y desencadenantes de la iteración; (2) Evaluar alternativas; Identificar y resolver los riesgos; (3) desarrollar y verificar los resultados de la iteración, y (4) plan de la próxima iteración.
- Cada ciclo comienza con la identificación de los interesados y sus condiciones de ganancia, y termina con la revisión y examinación.

2.2.4.5 *Rapid Application Development (RAD)*

El *desarrollo rápido de aplicaciones* (RAD) es una metodología de desarrollo de software, que implica el desarrollo interactivo y la construcción de prototipos. El desarrollo rápido de aplicaciones es un término originalmente utilizado para describir un proceso de desarrollo de software introducido por James Martin en 1991.

Principios básicos:

- Objetivo clave es para un rápido desarrollo y entrega de una alta calidad en un sistema de relativamente bajo coste de inversión.
- Intenta reducir los riesgos inherentes del proyecto partiéndolo en segmentos más pequeños y proporcionar más facilidad de cambio durante el proceso de desarrollo.
- Orientación dedicada a producir sistemas de alta calidad con rapidez, principalmente mediante el uso de iteración por prototipos (en cualquier etapa de desarrollo), promueve la participación de los usuarios y el uso de herramientas de desarrollo computarizadas. Estas herramientas pueden incluir constructores de Interfaz gráfica de usuario (GUI), Computer Aided Software Engineering (CASE) las herramientas, los sistemas de gestión de bases de datos (DBMS), lenguajes de programación de cuarta generación, generadores de código, y técnicas orientada a objetos.
- Hace especial hincapié en el cumplimiento de la necesidad comercial, mientras que la ingeniería tecnológica o la excelencia es de menor importancia.
- Control de proyecto implica el desarrollo de prioridades y la definición de los plazos de entrega. Si el proyecto empieza a aplazarse, se hace hincapié en la reducción de requisitos para el ajuste, no en el aumento de la fecha límite.
- En general incluye Joint application development (JAD), donde los usuarios están intensamente participando en el diseño del sistema, ya sea a través de la creación de consenso estructurado en talleres, o por vía electrónica.
- La participación activa de los usuarios es imprescindible.
- Iterativamente realiza la producción de software, en lugar de enfocarse en un prototipo.
- Produce la documentación necesaria para facilitar el futuro desarrollo y mantenimiento.

2.3 HERRAMIENTAS DE DESARROLLO DE SOFTWARE

Las **herramientas CASE** (*Computer Aided Software Engineering*, Ingeniería de Software Asistida por Computadora) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y de dinero. Estas herramientas nos pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costos, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras, que analizaba la relación existente entre los requisitos de un problema y las necesidades que éstos generaban, el lenguaje en cuestión se denominaba PSL (Problem Statement Language) y la aplicación que ayudaba a buscar las necesidades de los diseñadores PSA (Problem Statement Analyzer).

Aunque éstos son los inicios de las herramientas informáticas que ayudan a crear nuevos proyectos informáticos, la primera herramienta CASE fue Excelerator que salió a la luz en el año 1984 y trabajaba bajo una plataforma PC.

Las herramientas CASE alcanzaron su techo a principios de los años 90. En la época en la que IBM había conseguido una alianza con la empresa de software AD/Cycle para trabajar con sus mainframes, estos dos gigantes trabajaban con herramientas CASE que abarcaban todo el ciclo de vida del software. Pero poco a poco los mainframes han ido siendo menos utilizados y actualmente el mercado de las Big CASE ha muerto completamente abriendo el mercado de diversas herramientas más específicas para cada fase del ciclo de vida del software.

2.4 HIPÓTESIS

La implantación de un Sistema de Evaluación Curricular para el Bachillerato Informático del colegio Sultana del Oriente de la ciudad de Macas en entorno Web, aplicando la norma ISO/IEC 9216 incidirá en el cumplimiento del plan anual curricular.

2.5 IDENTIFICACIÓN DE VARIABLES

2.5.1 *VARIABLE INDEPENDIENTE*

- Sistema de Evaluación Curricular.

2.5.2 *VARIABLE DEPENDIENTE*

- Control del seguimiento en el cumplimiento del plan anual de contenidos.

CAPITULO III

PROCESO DE INGENIERIA DE SOFTWARE DEL SISTEMA

3.1 EVALUACIÓN CURRICULAR EN EDUCACIÓN.

Se presenta una propuesta metodológica para la evaluación del currículo en modelos a distancia, la cual será validada en la carrera de Informática del Colegio Particular a Distancia Sultana del Oriente en un proceso de autoevaluación.

Esta experiencia establece la necesidad de que los programas académicos inicien su proceso de mejoramiento continuo con el establecimiento de una línea base, realizando una evaluación en la que se diagnostica el estado de los planes de estudio y de cada uno de los cursos que lo integran. Posterior a esta evaluación la carrera se deberá iniciar un rediseño que dará como resultado un planteamiento curricular renovado que en adelante se mantendrá en cambio constante, sujeto al monitoreo a lo largo de la vida de la carrera.

Ante esta situación resulta necesario contar con una metodología y con instrumental para hacer una evaluación integral, un análisis riguroso de los aspectos cruciales del plan de estudios y una clara identificación de las verdaderas debilidades y fortalezas sobre las cuales desarrollar el proceso de mejoramiento continuo.

Como respuesta a esta necesidad se diseñará una metodología que permitirá a los responsables desarrollar un proceso participativo de evaluación del currículo y al mismo tiempo un monitoreo constante para evaluar el cumplimiento de metas y de los contenidos que se encuentran en cada malla.

El proceso de evaluación curricular se desarrollará mediante una metodología participativa que involucrará al personal académico, al estudiante, graduados, empleadores y expertos en diferentes momentos tanto de la revisión como juzgamiento de la calidad de los aspectos curriculares de la carrera de informática.

3.2 ENFOQUE TEÓRICO

3.2.1 *DEFINICIÓN DE CALIDAD DEL SOFTWARE*

La Calidad del software es el desarrollo de software basado en estándares con la funcionalidad y rendimiento total que satisfacen los requerimientos del cliente.

Además dice que es un proceso de desarrollo, gestión de proyectos, análisis y diseño, especificación de requerimientos, arquitectura, son solo algunos de los componentes que se aglomeran para conformar la ingeniería de software (IS) como disciplina para la creación y mantenimiento de software. Dentro de ésta, existe un subconjunto de teorías, herramientas y métodos orientados a lo que se denomina la calidad del software. Para resumir de alguna manera la amplitud de este concepto, se puede decir que la calidad de software ha sido usada desde un simple argumento de venta, hasta verdaderos estudios formales y usos de métricas para el desarrollo de software. Extrañamente dentro de la IS, la calidad del software es muy complicada de definir y de enmarcar en un simple concepto teórico.

Una idea general sobre un software de calidad es aquel que debiera cumplir con los requerimientos funcionales y de performance además de ser mantenible, confiable y aceptable.

3.2.2 *QUE SON LAS NORMAS ISO*

ISO es una federación mundial que agrupa a representantes de cada uno de los organismos nacionales de estandarización (como lo es el IRAM en la Argentina), y que tiene como objeto desarrollar estándares internacionales que faciliten el comercio internacional.

Cuando las organizaciones tienen una forma objetiva de evaluar la calidad de los procesos de un proveedor, el riesgo de hacer negocios con dicho proveedor se reduce en gran medida, y si los estándares de calidad son los mismos para todo el

mundo, el comercio entre empresas de diferentes países puede potenciarse en forma significativa

Durante las últimas décadas, organizaciones de todos los lugares del mundo se han estado preocupando cada vez más en satisfacer eficazmente las necesidades de sus clientes, pero las empresas no contaban, en general, con literatura sobre calidad que les indicara de qué forma, exactamente, podían alcanzar y mantener la calidad de sus productos y servicios.

De forma paralela, las tendencias crecientes del comercio entre naciones reforzaba la necesidad de contar con estándares universales de la calidad. Sin embargo, no existía una referencia estandarizada para que las organizaciones de todo el mundo pudieran demostrar sus prácticas de calidad o mejorar sus procesos de fabricación o de servicio.

Teniendo como base diferentes antecedentes sobre normas de estandarización que se fueron desarrollando principalmente en Gran Bretaña, la ISO creó y publicó en 1987 sus primeros estándares de dirección de la calidad: los estándares de calidad de la serie ISO 9000.

Con base en Ginebra, Suiza, esta organización ha sido desde entonces la encargada de desarrollar y publicar estándares voluntarios de calidad, facilitando así la coordinación y unificación de normas internacionales e incorporando la idea de que las prácticas pueden estandarizarse tanto para beneficiar a los productores como a los compradores de bienes y servicios. Particularmente, los estándares ISO 9000 han jugado y juegan un importante papel al promover un único estándar de calidad a nivel mundial.

3.2.3 LA FAMILIA ISO

Las series de normas ISO relacionadas con la calidad constituyen lo que se denomina familia de normas, las que abarcan distintos aspectos relacionados con la calidad:

ISO 9000: Sistemas de Gestión de Calidad Fundamentos, vocabulario, requisitos, elementos del sistema de calidad, calidad en diseño, fabricación, inspección, instalación, venta, servicio post venta, directrices para la mejora del desempeño.

ISO 10000: Guías para implementar Sistemas de Gestión de Calidad/ Reportes Técnicos Guía para planes de calidad, para la gestión de proyectos, para la documentación de los SGC, para la gestión de efectos económicos de la calidad, para aplicación de técnicas estadísticas en las Normas ISO 9000. Requisitos de aseguramiento de la calidad para equipamiento de medición, aseguramiento de la medición.

ISO 14000: Sistemas de Gestión Ambiental de las Organizaciones. Principios ambientales, etiquetado ambiental, ciclo de vida del producto, programas de revisión ambiental, auditorías.

3.2.4 COMO INFLUYEN UNA NORMA ISO EN EL DESARROLLO DE SOFTWARE.

ISO 9216 es un estándar internacional para la evaluación de la calidad del software. Está remplazado por el proyecto SQuaRE, ISO 25000:2005, el cual sigue los mismos conceptos.

Calidad Aplicación Web

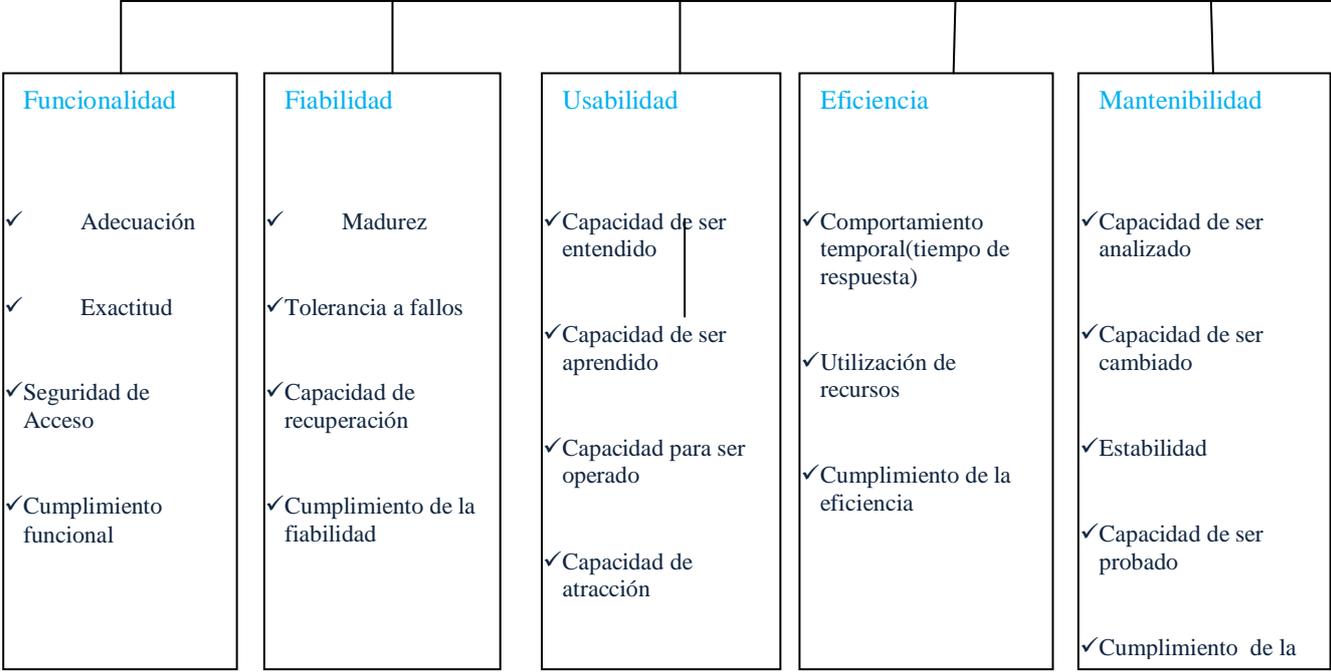


Ilustración 2 Parámetros de Calidad

El estándar está dividido en cuatro partes las cuales dirigen, respectivamente, lo siguiente: modelo de calidad, métricas externas, métricas internas y calidad en las métricas de uso.

El modelo de calidad establecido en la primera parte del estándar, ISO 9216-1, clasifica la calidad del software en un conjunto estructurado de características y subcaracterísticas de la siguiente manera:

- Funcionalidad - Un conjunto de atributos que se relacionan con la existencia de un conjunto de funciones y sus propiedades específicas. Las funciones son aquellas que satisfacen las necesidades implícitas o explícitas.
 - Idoneidad
 - Exactitud
 - Interoperabilidad
 - Seguridad
 - Cumplimiento de normas.

- Fiabilidad - Un conjunto de atributos relacionados con la capacidad del software de mantener su nivel de prestación bajo condiciones establecidas durante un período establecido.
 - Madurez
 - Recuperabilidad
 - Tolerancia a fallos

- Usabilidad - Un conjunto de atributos relacionados con el esfuerzo necesario para su uso, y en la valoración individual de tal uso, por un establecido o implicado conjunto de usuarios.
 - Aprendizaje
 - Comprensión
 - Operatividad

- Atractividad
- Eficiencia - Conjunto de atributos relacionados con la relación entre el nivel de desempeño del software y la cantidad de recursos necesitados bajo condiciones establecidas.
 - Comportamiento en el tiempo
 - Comportamiento de recursos
- Mantenibilidad - Conjunto de atributos relacionados con la facilidad de extender, modificar o corregir errores en un sistema software.
 - Estabilidad
 - Facilidad de análisis
 - Facilidad de cambio
 - Facilidad de pruebas
- Portabilidad - Conjunto de atributos relacionados con la capacidad de un sistema software para ser transferido desde una plataforma a otra.
 - Capacidad de instalación
 - Capacidad de reemplazamiento
 - Adaptabilidad
 - Co-Existencia

La subcaracterística Conformidad no está listada arriba ya que se aplica a todas las características. Ejemplos son conformidad a la legislación referente a usabilidad y fiabilidad.

Cada subcaracterística (como adaptabilidad) está dividida en atributos. Un atributo es una entidad la cual puede ser verificada o medida en el producto software. Los atributos no están definidos en el estándar, ya que varían entre diferentes productos software.

Un producto software está definido en un sentido amplio como: los ejecutables, código fuente, descripciones de arquitectura, y así. Como resultado, la noción de usuario se amplía tanto a operadores como a programadores, los cuales son usuarios de componentes como son bibliotecas software.

El estándar provee un entorno para que las organizaciones definan un modelo de calidad para el producto software. Haciendo esto así, sin embargo, se lleva a cada organización la tarea de especificar precisamente su propio modelo. Esto podría ser hecho, por ejemplo, especificando los objetivos para las métricas de calidad las cuales evalúan el grado de presencia de los atributos de calidad.

Métricas internas son aquellas que no dependen de la ejecución del software (medidas estáticas).

Métricas externas son aquellas aplicables al software en ejecución. La calidad en las métricas de uso están sólo disponibles cuando el producto final es usado en condiciones reales.

Idealmente, la calidad interna no necesariamente implica calidad externa y esta a su vez la calidad en el uso.

Este estándar proviene desde el modelo establecido en 1977 por McCall y sus colegas, los cuales propusieron un modelo para especificar la calidad del software. El modelo de calidad McCall está organizado sobre tres tipos de Características de Calidad:

- Factores (especificar): Describen la visión externa del software, como es visto por los usuarios.
- Criterios (construir): Describen la visión interna del software, como es visto por el desarrollador.
- Métricas (controlar): Se definen y se usan para proveer una escala y método para la medida.

ISO 9216 distingue entre fallo y no conformidad. Un fallo es el incumplimiento de los requisitos previos, mientras que la no conformidad es el incumplimiento de

los requisitos especificados. Una distinción similar es la que se establece entre validación y verificación.

3.3 HERRAMIENTAS Y LENGUAJES EMPLEADOS EN EL DESARROLLO DEL SISTEMA SULTANA

AppServ Es un paquete completo para la instalación de Apache, MySQL, PHP y phpMyAdmin sobre Microsoft Windows.

Características y Requisitos

Está desarrollado bajo la licencia [GNU/GPL], su instalación es muy sencilla, solo con ejecutar el instalador y en apenas un minuto tenemos disponibles un Servidor Web Apache, PHP, MySQL y phpMyAdmin.

Diferencia de la versión de AppServ

- 2.4.x es una obra de versión estable excelente para todo usuario, por la manera en que esta versión usa 4.x de PHP porque trabaje bien con sus scripts de PHP.
- 2.5.x es la función de roca. Esta versión provee el más nuevo y experimental de Apache, PHP, MySQL.
- 2.6.x incorpora la nueva versión de PHP 6.0.0 así como la versión 6.0.4 alpha de MySQL.

Aplicaciones

AppServ está diseñado con fines de aprendizaje sobre Windows no para utilizar como servidor Web profesional para lo cual no es recomendado.

¿Qué es PHP?

PHP (acrónimo de **PHP: HypertextPreprocessor**) es un lenguaje de código abierto muy popular especialmente adecuado para desarrollo web y que puede ser incrustado en HTML.

Una respuesta corta y concisa, pero, ¿qué significa realmente? Un ejemplo nos aclarará las cosas:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <title>Example</title>
  </head>
  <body>
    <?php
      echo "Hola, ¡soy un script PHP!";
    ?>
  </body>
</html>
```

Ejemplo 1 Un ejemplo introductorio

En lugar de usar muchos comandos para mostrar HTML (como en C o Perl), páginas PHP contienen HTML con código incluido en el mismo que hace "algo" (en este caso, mostrar "Hola ¡soy un script PHP!"). El código PHP está en medio de etiquetas de comienzo y final especiales `<?php` y `?>` que nos permitirán entrar y salir del "modo PHP".

Lo que distingue a PHP de algo como Javascript, es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá los resultados de ejecutar el script, sin ninguna posibilidad de determinar qué código ha producido el resultado recibido. El servidor web puede ser incluso configurado para que procese todos los archivos HTML con PHP y entonces no hay manera que los usuarios puedan saber que hay detrás de todo.

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. Aunque el desarrollo de PHP está centrado en programación de scripts en lado-servidor, se puede utilizar para muchas otras cosas.

Un concepto de php

Es un lenguaje para programar scripts del lado del servidor, que se incrustan dentro del código HTML. Este lenguaje es gratuito y multiplataforma.

PHP es el acrónimo de Hipertext Preprocesor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación.

Un lenguaje del lado del servidor es aquel que se ejecuta en el servidor web, justo antes de que se envíe la página a través de Internet al cliente. Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red, y otras tareas para crear la página final que verá el cliente. El cliente solamente recibe una página con el código HTML resultante de la ejecución de PHP. Como la página resultante contiene únicamente código HTML, es compatible con todos los navegadores

Con algo de conocimiento de programación de scripts del lado del servidor se puede hablar de PHP. PHP se escribe dentro del código HTML, lo que lo hace realmente fácil de utilizar, al igual que ocurre con el popular ASP de Microsoft, pero con algunas ventajas como su gratuidad, independencia de plataforma, rapidez y seguridad. Se puede descargar¹ de manera gratuita, un módulo que hace que nuestro servidor web comprenda los scripts realizados en este lenguaje. Es independiente de plataforma, puesto que existe un módulo de PHP para casi cualquier servidor web. Esto hace que cualquier sistema pueda ser compatible con el lenguaje y significa una ventaja importante, ya que permite portar el sitio desarrollado en PHP de un sistema a otro sin prácticamente ningún trabajo.

PHP, en el caso de estar montado sobre un servidor Linux u Unix, es más rápido que ASP, dado que se ejecuta en un único espacio de memoria y esto evita las comunicaciones entre componentes COM que se realizan entre todas las tecnologías implicadas en una página ASP.

Por último se señala la seguridad, en este punto también es importante el hecho de que en muchas ocasiones PHP se encuentra instalado sobre servidores Unix o

¹ www.php.net

Linux, que son conocidos como más veloces y seguros que el sistema operativo donde se ejecuta las ASP, Windows NT o 2000. Además, PHP permite configurar el servidor de modo que se permita o rechacen diferentes usos, lo que puede hacer al lenguaje más o menos seguro dependiendo de las necesidades de cada cual.

Este lenguaje de programación está preparado para realizar muchos tipos de aplicaciones web gracias a la extensa librería de funciones con las que está dotado. La librería de funciones cubre desde cálculos matemáticos complejos hasta tratamiento de conexiones de red.

Algunas de las más importantes capacidades de PHP son:

- **Compatibilidad con las bases de datos** más comunes, como MySQL, mSQL, Oracle, Informix, y ODBC, por ejemplo.
- Incluye funciones para el **envío** de correo electrónico, upload de archivos, crear dinámicamente en el servidor imágenes en formato GIF, incluso animadas y una lista interminable de utilidades adicionales.

CSS:

Es un término muy familiar entre desarrolladores y webmasters, aunque algunos desconocen su significado y utilidad. Entender la dinámica del CSS es algo esencial entre los desarrolladores web y parte de sus obligaciones en su tarea diaria. Aprender CSS es una tarea que exige constancia y práctica.

CSS es un lenguaje como otro cualquiera orientado a definir los parámetros esenciales en un documento HTML. Por ejemplo, con CSS customizamos parámetros acerca de las fuentes, colores, márgenes, líneas, imágenes de fondo, posición de contenidos, tabulaciones, anchuras, rellenos (padding), etc.

HTML permite desarrollar un **documento básico pre formato**. Con CSS podemos definir todo el formato de una manera sofisticada y personalizada. CSS es el código que leen todos los navegadores actuales, aunque cada uno a su debida manera. Los navegadores hacen interpretaciones del CSS y recrean su lectura en el sitio web donde se procesa.

HTML es utilizado para estructurar los contenidos, es el cuerpo que contiene la información. Hace algún tiempo Tim Berners Lee decidió hacer la inventiva de la World Wide Web (WWW), y el lenguaje HTML servía para estructurar los textos. La importancia de los textos era definida utilizando etiquetas como <h1>, <h2>, y <h3> que se conocen como “headings” (encabezados).

Con la popularidad de la Web, los diseñadores comenzaron a sentir la necesidad de construir documentos con un formato más personalizado. Para atender a esta necesidad, los fabricantes de navegadores inventaron nuevas etiquetas como que pasó a ser empleado como modo de definir un texto.

Muchas de estas etiquetas eran soportadas solamente por algunos navegadores. Como solución se puso a disposición el CSS.

¿Qué es el CSS?

CSS es una revolución en el mundo del diseño que incluye innumerables beneficios. Gracias a él, se ha potenciado la creatividad de los diseñadores desde su origen.

- Control de varios documentos a partir de un mismo fichero CSS.
- Mayor precisión y facilidad de edición ya que todos se vinculan al mismo fichero patrón.
- Aplicación de diferentes “layouts” para servir a diferentes medidas: resoluciones de pantalla, impresora, soportes digitales, etc.
- Empleo de sofisticadas técnicas de desarrollo.

¿Qué es java script?

JavaScript es el lenguaje de programación más popular en internet y trabaja en todos los navegadores más importantes, tales como Internet Explorer, Firefox, cromo, Opera y Safari.

- JavaScript fue diseñado para añadir interactividad a páginas HTML
- JavaScript es un lenguaje de secuencias de comandos
- Un lenguaje de secuencias de comandos es un lenguaje de programación ligero
- JavaScript está normalmente incrustada directamente en las páginas HTML
- JavaScript es un lenguaje interpretado (significa que puede ejecutar secuencias de comandos sin necesidad de compilación)
- Cualquier persona puede utilizar JavaScript sin necesidad de adquirir una licencia

¿Son Java y JavaScript lo mismo?

- Java y JavaScript son dos lenguajes completamente diferentes en concepto y diseño!
- Java (desarrollado por Sun Microsystems) es un potente y mucho más complejo lenguaje de programación - en la misma categoría como c y C++.

¿Qué puede hacer un JavaScript?

JavaScript brinda a los diseñadores HTML una herramienta de programación. Los diseñadores HTML normalmente no son programadores, pero JavaScript es un lenguaje con una sintaxis muy simple.

JavaScript puede poner texto dinámico en una página HTML, -Un comando de JavaScript como: `document.write("<h1>" + nombre + "</h1>")` puede escribir un texto o variable en una página HTML.

JavaScript puede reaccionar a eventos, -Un JavaScript se puede programar para ejecutarse cuando sucede algo, como cuando ha terminado de cargar una página o cuando un usuario hace clic en un elemento HTML.

Puede leer y escribir elementos HTML. Un JavaScript puede leer y modificar el contenido de un elemento HTML. Puede utilizar JavaScript para validar datos.

JavaScript puede utilizarse para detectar el navegador del visitante y dependiendo del navegador cargar otra página diseñada específicamente para ese navegador. JavaScript puede utilizarse para crear cookies: JavaScript puede utilizarse para almacenar y recuperar información en equipo del visitante

Algo de Historia sobre JavaScript.

JavaScript es una implementación del lenguaje ECMAScript estándar, ECMAScript es desarrollado y mantenido por la Organización de ECMA.

ECMA-262 es el estándar de JavaScript oficial.

El lenguaje fue inventado por Brendan Eich de Netscape (con Navigator 2.0) y ha aparecido en todos los navegadores Netscape y Microsoft desde 1996.

El desarrollo de ECMA-262 comenzó en 1996, y la primera edición fue adoptada por la Asamblea General de ECMA en junio de 1997.

El estándar fue aprobado como un internacional ISO (ISO/IEC 16262) estándar en 1998.

Base de datos MySQL

MySQL es el servidor de bases de datos relacionales más popular, desarrollado y proporcionado por MySQL AB. MySQL AB es una empresa cuyo negocio consiste en proporcionar servicios en torno al servidor de bases de datos MySQL. Se puede obtener también como una biblioteca multihilo que se puede enlazar dentro de otras aplicaciones para obtener un producto más pequeño, más rápido, y más fácil de manejar. MySQL es un sistema de administración de bases de datos.

Una base de datos es una colección estructurada de datos. La información que puede almacenar una base de datos puede ser tan simple como la de una agenda, un contador, o un libro de visitas, o tan vasta como la de una tienda en línea, un sistema de noticias, un portal, o la información generada en una red corporativa. Para agregar, acceder, y procesar los datos almacenados en una base de datos, se necesita un sistema de administración de bases de datos, tal como MySQL.

MySQL es un sistema de administración de bases de datos relacionales. Una base de datos relacional almacena los datos en tablas separadas en lugar de poner todos los datos en un solo lugar. Esto agrega velocidad y flexibilidad. Las tablas son enlazadas al definir relaciones que hacen posible combinar datos de varias tablas cuando se necesitan consultar datos. La parte SQL de "MySQL" significa "Lenguaje Estructurado de Consulta", y es el lenguaje más usado y estandarizado para acceder a bases de datos relacionales.

MySQL es Open Source

Open Source significa que la persona que quiera puede usar y modificar MySQL. Cualquiera puede descargar el software de MySQL de Internet y usarlo sin pagar por ello. Inclusive, cualquiera que lo necesite puede estudiar el código fuente y cambiarlo de acuerdo a sus necesidades. MySQL usa la licencia GPL (Licencia Pública General GNU), para definir qué es lo que se puede y no se puede hacer

con el software para diferentes situaciones. Sin embargo, si uno está incómodo con la licencia GPL o tiene la necesidad de incorporar código de MySQL en una aplicación comercial es posible comprar una versión de MySQL con una licencia comercial.

¿Por qué usar MySQL?

El servidor de bases de datos MySQL es muy rápido, seguro, y fácil de usar. Se pueden encontrar comparaciones de desempeño con algunos otros manejadores de bases de datos en la página de MySQL.

El servidor MySQL fue desarrollado originalmente para manejar grandes bases de datos mucho más rápido que las soluciones existentes y ha estado siendo usado exitosamente en ambientes de producción sumamente exigentes por varios años. Aunque se encuentra en desarrollo constante, el servidor MySQL ofrece hoy un conjunto rico y útil de funciones. Su conectividad, velocidad, y seguridad hacen de MySQL un servidor bastante apropiado para acceder a bases de datos en Internet.

El software de bases de datos MySQL consiste de un sistema cliente/servidor que se compone de un servidor SQL multihilo, varios programas clientes y bibliotecas, herramientas administrativas, y una gran variedad de interfaces de programación (APIs).

Dreamweaver

Adobe Dreamweaver es una aplicación en forma de estudio (basada en la forma de estudio de Adobe Flash) enfocada a la construcción y edición de sitios y aplicaciones Web basados en estándares. Creado inicialmente por Macromedia (actualmente producido por Adobe Systems). Es el programa de este tipo más utilizado en el sector del diseño y la programación web, por sus funcionalidades, su integración con otras herramientas como Adobe Flash y, recientemente, por su soporte de los estándares del World Wide Web Consortium. Su principal competidor es Microsoft Expression Web y tiene soporte tanto para edición de imágenes como para animación a través de su integración con otras. Hasta la versión MX, fue duramente criticado por su escaso soporte de los estándares de la web, ya que el código que generaba era con frecuencia sólo válido para Internet Explorer, y no validaba como HTML estándar. Esto se ha ido corrigiendo en las versiones recientes.

Adobe Dreamweaver permite crear sitios de forma totalmente gráfica, y dispone de funciones para acceder al código HTML generado. Permite la conexión a un servidor, a base de datos, soporte para programación en ASP, PHP, Javascript, cliente FTP integrado, etc.

NOTEPAD++

Notepad++ es un editor de texto y de código fuente libre con soporte para varios lenguajes de programación. Solo funciona en Microsoft Windows.

Se parece al Bloc de notas en cuanto al hecho de que puede editar texto sin formato y de forma simple. No obstante, incluye opciones más avanzadas que pueden ser útiles para usuarios avanzados como desarrolladores y programadores.

Se distribuye bajo los términos de la Licencia Pública General de GNU.

Características

- **Coloreado y envoltura de sintaxis:** si se escribe en un lenguaje de programación o marcado, Notepad++ es capaz de resaltar las expresiones propias de la sintaxis de ese lenguaje para facilitar su lectura.
- **Pestañas:** al igual que en muchos navegadores, se pueden abrir varios documentos y organizarlos en pestañas.
- **Resaltado de paréntesis e indentación:** cuando el usuario coloca el cursor en un paréntesis, Notepad++ resalta éste y el paréntesis correspondiente de cierre o apertura. También funciona con corchetes y llaves
- **Grabación y reproducción de macros.**
- **Soporte de extensiones:** incluye algunas por defecto.

Lenguajes soportados

Ada	Makefile
ASP	MATLAB
ASM Ensamblador (x86)	MS INI (archivo)
ASM Ensamblador (Z80)	NSIS
AutoIt	Objective-C
Batch	Pascal
C	Perl
C#	PHP
C++	PostScript
Caml	PowerShell
CMake	Properties
COBOL	Python
CSS	R
D	RC (fichero de recurso)
Diff	Ruby
Flash ActionScript	Shell
Fortran	Scheme
Gui4Cli	Smalltalk
Haskell	SQL
HTML	Tcl
INNO	TeX
Java	VB
JavaScript	VHDL
JSP	Verilog

KiXtart	XML
Lisp	YAML
Lua	

Tabla 1 Lenguajes soportados por Notepad++

Además, permite al usuario definir su propio lenguaje, no sólo las palabras clave para la sintaxis coloreada, sino también las palabras clave para la envoltura de sintaxis, los comentarios clave y los operadores.

Está basado en el componente de edición Scintilla y está escrito en C++ utilizando directamente la API de Windows y STL, lo que asegura una velocidad mayor de ejecución y un tamaño más reducido del programa final.

PhpMyAdmin

Introducción.- PhpMyAdmin es una herramienta de software libre escrito en PHP para ocuparse de la administración de MySQL sobre la Red en el ámbito mundial. phpMyAdmin es compatible con una amplia gama de operaciones con MySQL. La mayoría de las operaciones de uso frecuente son compatibles con la interfaz de usuario (administrar bases de datos, tablas, campos, relaciones, índices, usuarios, permisos, etc), mientras se puede ejecutar cualquier sentencia SQL directamente.

Definición. PhpMyAdmin es una utilidad que sirve para interactuar con una base de datos de forma muy sencilla y desde una interfaz web. Nos sirve por ejemplo para crear bases de datos, tablas, borrar o modificar datos, añadir registros, hacer copias de seguridad, etc. Es una aplicación tan útil que casi todos los hosting con MySQL disponen de ella. Además, se usa para crear los usuarios MySQL para así poder utilizar las bases de datos de forma segura. Al ser una aplicación escrita en PHP, necesita de Apache y MySQL para poder funcionar. phpMyAdmin es también muy profundamente documentados en un libro escrito por uno de los desarrolladores. Dominar phpMyAdmin para una eficaz gestión de MySQL , que está disponible en Inglés, checo, alemán y español. Para facilitar el uso de una amplia gama de personas, phpMyAdmin es traducido a 58 idiomas.

Características: interfaz web intuitiva Soporte para las características más MySQL: navegar y borrar bases de datos, tablas, vistas, campos e índices crear, copiar, eliminar y alterar cambiar el nombre de bases de datos, tablas, campos e índices mantenimiento del servidor, bases de datos y tablas, con propuestas sobre la configuración del servidor ejecutar, editar y marcar cualquier sentencia SQL, incluso por lotes consultas administrar usuarios y privilegios de MySQL gestión de procedimientos almacenados y disparadores Importar datos de CSV y SQL Exportación de datos a varios formatos: CSV, SQL, XML, PDF, ISO / IEC 26300 - OpenDocument Text y hoja de cálculo, de Word, Excel, A L T E X y otros La administración de múltiples servidores Creación de gráficos en PDF de su base de datos de diseño Creación de consultas complejas utilizando Consulta por ejemplo (QBE) Búsqueda a nivel mundial en una base de datos o un subconjunto de la misma La transformación de los datos almacenados en cualquier formato utilizando un conjunto de funciones predefinidas, como mostrar los datos BLOB como imagen o enlace de descarga.

Requisitos: Es necesario contar con un servidor web con soporte de PHP, un motor de bases de datos MySQL y, por supuesto, un navegador web.

INSTALACION: La herramienta se descarga desde sitio web oficial². Una vez descargado, hay que descomprimirlo en el directorio raíz del servidor web. Se generará la carpeta phpMyAdmin quedando los directorios de la siguiente manera: c:\inetpub\wwwroot\phpMyAdmin. Si se desea, puede renombrar la carpeta root\phpMyAdmin a phpMyAdmin, se puede modificar a conveniencia.

Configuración: Abrimos el archivo config.inc.php con algún editor de texto como Edit Plus o simplemente el Notepad de Windows. Debemos configurar las siguientes instancias: `$cfg['PmaAbsoluteUri'] = 'http://localhost/phpMyAdmin';` Hay que asignar la ruta completa para acceder al phpMyAdmin. Si trabajamos en forma local: `http://localhost/phpMyAdmin` y en caso de trabajar en un servidor `http://www.servidor.com/phpMyAdmin.` `$cfg['Servers']`

² : www.phpmyadmin.net.

[`$i`]['host'] = 'localhost'; El host o número de IP del servidor de la base de datos MySQL. `cfg['Servers'][$i]['user']` = 'usuario'; El nombre del usuario de la base de datos. Si trabajamos en forma local generalmente pondremos el usuario root (súper usuario) para poder administrar usuarios y bases de datos desde el navegador. En caso de utilizar un servidor externo, seguramente sólo tendremos privilegios de usuario común con una sola base de datos.

`$cfg['Servers'][$i]['password']`='contraseña'; La contraseña del usuario especificado en la línea anterior.

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

SU INSTALACION: Bajarse el PHP: La página principal de PHP³. Una vez bajado hay que descomprimirlo; una vez descomprimido en c: tendremos una carpeta que se llame c:php-4.2.3-Win32, lo mejor es cambiarle el nombre, la llamaremos c:php a secas. Ahora en una carpeta que se llama c:php con el contenido de php; lo siguiente que tenemos que hacer es entrar en esa carpeta y copiar el archivo php.ini-dist a nuestro directorio de Windows (normalmente c:windows) con el nombre php.ini; el archivo debe quedar así: c:windowsphp.ini

Modificaciones en Apache para que soporte PHP Tenemos que abrir el archivo httpd.conf (se encuentra en c:apacheapache2conf) y añadir al final del mismo este texto: `ScriptAlias /php/ "c:/php/"`; `AddTypeapplication/x-httpd-php .php``Actionapplication/x-httpd-php"/php/php.exe"`; Si tenemos iniciado

³ <http://www.php.net>

nuestro servidor Apache, lo tenemos que reiniciar para que soporte PHP, ahora mismo funcionaría PHP en el servidor web.

Modificaciones en el php.ini: Abrir el archivo php.ini (está en c:windowsphp.ini) y buscar el texto que pone: ; You should do your best to write your scripts so that they do not require ; register_ to be on; Using form variables as globals can easily lead ; to possible security problems, if the code is not very well thought of. register_globals=Off. Para que cuando se vaya a usar los métodos POST y GET funcionen correctamente es necesario poner: register_globals = On. La segunda y última modificación que se hace sobre el php.ini es esta; buscar este texto: [mail function] ; For Win32 only.

CAPITULO IV

DESARROLLO DEL SISTEMA

4.1 IMPLANTACION DEL SISTEMA DE EVALUACIÓN CURRICULAR

4.1.1 DEFINICIÓN DEL PROBLEMA

Debido a las reformas educativas lanzadas por el actual gobierno y, en particular, la evaluación de un plantel educativo por medio del resultado de la evaluación de los alumnos de los terceros años del bachillerato mediante el examen de ingreso a las universidades del país, hace imprescindible un sistema automatizado que permita conocer y evaluar el cumplimiento riguroso del plan académico.

En la actualidad el Colegio Sultana del Oriente no posee un software que mejore la automatización de sus procesos académicos, lo que ocasiona gran pérdida de tiempo en la realización de actividades repetitivas y constantes en cada año lectivo.

No hay una base de datos con los profesores que son parte de la institución, también se planifica anualmente las materias a ser impartidas por los mismos a cada curso, detallando los horarios y los maestros involucrados, pero no existe un reporte si se cumplió o no los objetivos que permita mejorar la calidad de educación en esta institución.

No existe estándar que mida la calidad de software desarrollo de software ágil que nos permita obtener una aplicación satisfactoria que cumpla con los requerimientos impuestos en el menor tiempo posible se vuelve una tarea complicada y en su mayoría no cumplen con ayudarnos a proporcionar aplicaciones de calidad.

4.1.2 DEFINIR ALTERNATIVAS DE SOLUCIÓN

Surge la necesidad de realizar un Sistema Informático que permita tener un seguimiento claro del nivel de cumplimiento de la malla curricular en los años de bachillerato que permita a los directivos tomar decisiones en la malla que se

impartirá a los estudiantes durante el año lectivo, además se lograra de esta manera optimizar tiempo, recursos, costo y pérdida de información.

Es por esto que es necesario, realizar una Aplicación Web que específicamente lo manejen los estudiantes, maestros y personal administrativo del Colegio particular a Distancia Sultana del Oriente, y que toda la comunidad educativa pueda acceder al sitio Web mediante un enlace.

4.1.3 DEFINICIÓN DEL ÁMBITO DEL SOFTWARE

El sistema se orienta exclusivamente al seguimiento y evaluación curricular del Bachillerato en Informática del tercer año de bachillerato informático del Colegio Sultana del Oriente de la ciudad de Macas.

Para tener un panorama claro del desarrollo del proyecto se hace necesario tomar muestras representativas por medio de encuestas a los estudiantes, autoridades y profesores para identificar los requerimientos de la aplicación.

Adicionalmente se trata de evaluar el cumplimiento de la malla curricular por parte de los alumnos y docentes del tercer año de bachillerato informático de la institución educativa.

4.2 DESCRIPCIÓN GENERAL

4.2.1 PERSPECTIVA DEL PRODUCTO

La Aplicación Web de Evaluación Curricular permitirá dar seguimiento e identificar los problemas en el desarrollo del ciclo diversificado y el cumplimiento de la malla curricular para de esta manera ir mejorando el desempeño de los alumnos, principalmente los que aspiren seguir una carrera universitaria.

4.3 DESCRIPCIÓN DEL COLEGIO SULTANA DEL ORIENTE

4.3.1 MISIÓN INSTITUCIONAL⁴

" El COLEGIO SULTANA DEL ORIENTE de la ciudad de MACAS, se empeña en el desarrollo educativo de los estudiantes utilizando la tecnología universal con aplicaciones propias y reales de acuerdo al medio en que se desarrolla, tomando en cuenta la problemática, oportunidades, fortalezas y amenazas que se presentan independientemente en cada sector o población, contribuyendo a la optimización de todo tipo de actividades, desarrollando así la capacidad para resolver problemas y crear como se menciona anteriormente nuestra propia cultura informática, nuestra misión es insertar ya a nuestros estudiantes una educación que integre la tecnología como apoyo y esta a su vez se inserte verdaderamente a nuestra vida para obtener beneficios.."

4.3.2 VISIÓN INSTITUCIONAL⁵

" Los cambios educativos a futuro, previstos por el COLEGIO SULTANA DEL ORIENTE DE LA CIUDAD DE MACAS - PROVINCIA DE MORONA SANTIAGO, se inspiran en el desarrollo y crecimiento personal, familiar, social, regional, nacional e internacional, todos entrelazados con valores como el afecto, punto imprescindible ya que el querernos a nosotros mismos, que se lo conoce como AUTOESTIMA, logra insertar puntos favorables y positivos dentro de los aspectos que se relacionan al desarrollo descentralizado, en cualquier actividad. Del avance tecnológico que envuelve a nuestro planeta, surge la necesidad de una educación técnica y tecnológica, la masificación de tecnologías, las ofertas existentes que invaden los mercados nos muestran un concepto errado, se PRETENDE QUE NOSOTROS NOS ACOMODEMOS A LAS TECNOLOGIAS EXTERNAS, QUE ASIMILEMOS LO QUE YA ESTA ESTRUCTURADO, ADAPTADO A OTRA REALIDAD, nos encontramos frente a un mercado y un mundo DOMINADO POR LA COMPETENCIA GLOBALIZADA. Debemos emprender nuestro camino utilizando la tecnología con nuestros

⁴ Tomado del folleto informativo del Colegio Sultana del Oriente

⁵ Tomado del folleto informativo del Colegio Sultana del Oriente

conceptos y aplicaciones a nuestras necesidades, características para generar así NUESTRA CULTURA INFORMATICA, TECNOLOGICA Y CIENTIFICA para poder sobrevivir en el actual y cambiante mercado."

4.3.3 CREACIÓN⁶

“El Colegio Técnico Particular a Distancia "SULTANA DEL ORIENTE" requiere obtener la resolución ministerial que le permite su legal funcionamiento y su consecuente oferta de servicios a la juventud estudiosa de MACAS.

Con la base legal que se obtenga por parte del Ministerio de Educación y Culturas del país, El Colegio Técnico Particular a Distancia "SULTANA DEL ORIENTE" estará destinado a la formación de bachilleres en las ramas de la producción de bienes y servicios.

La fundamentación de los tres años de la Educación Básica y la selección de los bachilleratos para el Colegio Técnico Particular a Distancia "SULTANA DEL ORIENTE" se promueve en correspondencia con los requerimientos del desarrollo socio económico del sector y se convierte en antecedentes importantes que constituyen un apreciable recurso referencial que permite afianzar el desarrollo institucional como un proceso pedagógico probado, calificado y satisfactoriamente aceptado por el país durante el tiempo que viene prestando los servicios educacionales.

El Proyecto para los bachilleres es un producto de un proceso de reflexión basadas en las necesidades de la ciudad, de la provincia, región y país en general, que han coadyuvado para recibir una propuesta educativa con la dinámica socioeconómica que enfrenta el reto de la promoción de las entidades privadas en el aporte al desarrollo del Ecuador.

Nuestra sociedad requiere del aporte de una institución con apertura técnica, con espíritu amplio y a tono con los requerimientos de la cultura, tecnología, civilización y ciencia actuales, que ofrezca su colaboración y ayuda. Esto hace evidente que los bachilleratos que se han planificado en el Colegio tenga como objetivo fundamental la actualización científica en las ramas tecnológicas y computacionales; y en las áreas humanísticas, de manera que no se prepare solo

⁶ Tomado del folleto informativo del Colegio Sultana del Oriente

bachilleres, sino bachilleres pragmáticos en los campos científicos a tono con los requerimientos del siglo XXI".

4.3.4 *MODELO PEDAGÓGICO*

El modelo pedagógico del CSO está basado en:

- Profesores ampliamente capacitados.
- Nuevas tecnologías al servicio de la educación.
- Uso del inglés como lenguaje universal.
- Atención personalizada a los alumnos.
- Proceso de enseñanza-aprendizaje activo y crítico.
- Formación integral.
- Actividades complementarias.

4.3.5 *GESTION ADMINISTRATIVA*

La complejidad de los procesos que maneja una institución educativa, conlleva a emplear gran cantidad de tiempo y dinero en la realización de tareas repetitivas, lo que implica una baja competitividad del personal de la institución en los servicios que brindan a sus estudiantes, ya que se dedican a tareas manuales que implica la utilización de gran cantidad de recursos y es susceptible a pérdida de información y errores humanos.

Sin un sistema que permita automatizar los procesos para una evaluación curricular adecuada en el Colegio Sultana del Oriente, se dificulta la mejor atención interna al personal docente y administrativo, y externa a los estudiantes y padres de familia, ya que no existe una interacción completa de todas las personas que participan en la formación de los estudiantes, sin tener la información disponible en el momento que esta se requiere para monitorear constantemente los resultados del avance curricular en las diferentes asignaturas.

La falta de integración de los procesos que se manejan en las diferentes áreas para la evaluación en el colegio, no permiten tener un completo control, que ayude a

incrementar la competitividad del área académica de la institución, ya que al no tener sus procesos complementados con tecnología, no dan a sus estudiantes un valor agregado, que integre la calidad académica que el colegio les brinda, con la utilización de tecnología acorde con las necesidades de la institución.

Actualmente el CSO no posee un sistema que automatice sus tareas evaluativas, lo que ocasiona que cada año lectivo la planificación académica se vuelva una tarea sumamente compleja, que frecuentemente ocasiona gran pérdida de tiempo y esfuerzo, desde inscripción de los alumnos en cada año lectivo, registro de notas, seguido por las dificultades que involucran manejar un amplio número de registros en hojas de cálculo, junto a la generación de reportes que ocupa gran cantidad de recursos materiales y humanos, que podrían ser canalizados en otras actividades.

4.3.6 GESTIÓN ACADÉMICA

La gestión estudiantil considera la importancia fundamental que tienen los estudiantes en toda institución educativa en el proceso de enseñanza aprendizaje. Es por esto que gran parte del servicio que se preste a los estudiantes y padres de familia dependerá de la manera en la que se administre los aspectos esenciales que intervienen en una gestión académica integral, tal como la información personal de los estudiantes, sus calificaciones de cada período escolar, y un correcto control en el seguimiento curricular.

Sin duda, entre los principales problemas que el Colegio Sultana del Oriente mantiene en su gestión estudiantil es la falta de un medio apropiado de seguimiento al cumplimiento curricular en el transcurso y al final del período lectivo.

Esto ha ocasionado en más de una vez, grandes inconvenientes causados por la desorganización y confusión creada por la falta de conocimiento integral de todo lo que abarca la institución educativa.

4.4 SELECCIÓN Y JUSTIFICACION DE LA METODOLOGÍA

Los proyectos de desarrollo de software de la actualidad se desenvuelven en un entorno muy cambiante donde se exige reducir los tiempos de desarrollo, manteniendo una alta calidad en su resultado final sin involucrar un aumento considerable en la complejidad de las fases en las que el proyecto es realizado. La elección de una metodología adecuada que se ajuste al equipo de desarrollo permitirá una flexibilidad para la obtención de los objetivos definidos al inicio del proyecto y además proveerá satisfacción al cliente más allá de sus necesidades o expectativas.

Dentro de la norma ISO/IEC 9216 concerniente a la calidad del software existen parámetros vistos en el capítulo anterior que rigen en el desarrollo de software para que un sistema pueda cumplir con dicha norma. De esta manera es muy importante escoger una metodología adecuada para alcanzar estos parámetros de calidad.

Ya que las metodologías ágiles están orientadas para proyectos pequeños que requieran un tiempo corto para desarrollo, se las consideran como una solución a la medida que no abandona las prácticas tradicionales que aseguran la calidad del producto final que se entrega al usuario.

Las metodologías de desarrollo ágil de software se basan en el Manifiesto del Desarrollo Ágil de Software, el cual pretende valorar:

*“Individuos e interacciones sobre procesos y herramientas
 Software que funciona sobre documentación exhaustiva
 Colaboración con el cliente sobre negociación de contratos
 Responder ante el cambio sobre seguimiento de un plan”⁷*

Se pueden apreciar las diferencias esenciales que entre las metodologías tradicionales versus las metodologías ágiles el siguiente cuadro comparativo.

Metodologías Ágiles	Metodologías Tradicionales
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Especialmente preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente (por el equipo de desarrollo)	Impuestas externamente
Proceso menos controlado, con pocos Principios	Proceso mucho más controlado, con numerosas políticas/normas
No existe contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (menor a 10 integrantes) y trabajando en el mismo sitio	Grupos grandes y posiblemente distribuidos
Pocos roles	Más roles
Menos énfasis en la arquitectura del Software	La arquitectura del software es esencial y se expresa mediante modelos

Tabla 2 Metodologías ágiles vs. Metodologías tradicionales⁸

⁷ <http://agilemanifesto.org/>

⁸ <http://www.willydev.net/descargas/masyxp.pdf>

4.4.1 COMPARATIVA ENTRE METODOLOGÍAS DE DESARROLLO

Las metodologías que han sido seleccionadas para la comparación se relacionan en alto grado con el tipo de desarrollo del presente proyecto, definiendo lineamientos a seguir relacionados al desarrollo ágil y web.

Entonces, se comparan a continuación XP (Programación Extrema), HDM (Método de Diseño Hipermedia), EORM (Metodología de Relaciones de Objetos Mejorada), OOHDM (Metodología de Diseño de Hipermedia Orientada a Objetos), SOHDM (Metodología de Diseño Hipermedia orientada a objetos y basada en escenarios), y WSDM (Método de Diseño de Sitios Web).

	PROCESOS	Técnica de modelado	Notación	Herramienta de soporte
XP	1. Planificación 2. Gestión 3. Diseño 4. Desarrollo 5. Pruebas	UML según se requiera	Flujogramas Diagramas de clase Diagramas de interacción Bocetos de IU, etc.	No especificada
HDM	1. Desarrollo a largo plazo 2. Desarrollo a corto plazo	Entidad-Relación	Diagrama E-R	No especificada
EORM	1. Clases del entorno de desarrollo. 2. Composición del entorno de desarrollo. 3. Entorno de desarrollo de UI.	Orientando a Objetos	1. Diagrama de clases 2. Diseño GU	ONTOS Studio
OOHDM	1. Diseño conceptual 2. Diseño navegacional 3. Diseño abstracto de la UI 4. Implementación	Orientando a Objetos	1. Diagrama de clases 2. Diagrama navegacional, clase + contexto. 3. Diagrama de configuración de ADV + Diagrama ADV (Abstract DataView)	OOHDM-Web
SOHDM	1. Análisis del dominio 2. Modelo en OO 3. Diseño de la vista 4. Diseño navegacional 5. Diseño implementación 6. Construcción	Escenarios, Vistas Orientando a Objetos	1. Diagramas de escenarios de actividad. 2. Diagrama de estructura de clase. 3. Vista OO. 4. Esquema de enlace navegacional. 5. Esquema de páginas.	No especificada
WSDM	1. Modelado del usuario 2. Diseño conceptual 2.1. Modelo objetos 2.2. Diseño navegacional 3. Diseño implementación 4. Implementación	Entidad-Relación Orientando a Objetos	1. Diagrama de E-R o clase. 2. Capas de navegación.	No especificada

Tabla 3 Comparación de metodologías⁹

4.4.2 CONCLUSIONES PARA SELECCIÓN DE METODOLOGÍA

⁹ http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22_art5_c.pdf

Una vez realizado el análisis comparativo entre las metodologías de desarrollo, se ha escogido Extreme Programming como determinante para orientar el proyecto, luego de haber llegado a las siguientes conclusiones:

- Debido a que el sistema del presente proyecto requiere entregas a corto plazo, cumpliendo con los requerimientos planteados, la metodología XP sugiere iteraciones que pueden aplicarse plenamente, llegando en cada una de ellas a satisfacer las necesidades existentes en la institución, tales como la inscripción de alumnos, manejo de matrículas, informes de notas a los alumnos, gestión de docentes y su asignación a cada curso y el motivo de este proyecto, dar seguimiento al cumplimiento de la malla curricular de cada período académico.
- El equipo de desarrollo necesario para aplicar XP es pequeño, apostando por el desarrollo modular del sistema y a reuniones constantes para refinar el sistema, basándose en la retroalimentación que se realiza conjuntamente con el cliente y en las pruebas unitarias; además de las normativas curriculares dispuestas por el Ministerio de Educación como por ejemplo la disposición de cada año dividido por quimestres en los cuales habrán dos aportaciones y un examen quimestral en fechas definidas, las mismas que deben cumplirse en los periodos determinados.
- Esta metodología plantea la reducción de tiempos de desarrollo en base a simplificar el diseño, disminuir el número de procesos involucrados; todo esto para agilizar los cambios que puedan darse en el transcurso del proyecto.

4.4.3 SELECCIÓN Y JUSTIFICACIÓN DE LAS HERRAMIENTAS DE DESARROLLO

El alto costo de licenciamiento y la poca flexibilidad que presenta el software propietario ha dado impulso a la utilización de software libre. Sin duda esta tendencia cada vez se robustece más debido a la libertad que los desarrolladores tenemos para modificar, personalizar y mejorarlo de acuerdo a las necesidades de cada institución, además el soporte se encuentra basado en la colaboración en la red.

Para la realización del Sistema de Evaluación Curricular para el Colegio Sultana del oriente es necesario contar con herramientas tales como servidor de

aplicaciones web, gestor de base de datos, lenguaje de programación y un Entorno de Desarrollo Integrado (IDE), las cuales han sido seleccionadas y se detallan a continuación.

4.4.4 *SERVIDOR DE APLICACIONES WEB*

Como servidor de aplicaciones web se decidió trabajar con Apache, debido a la experiencia con la que contamos para el manejo de esta herramienta es amplia, además se la ha utilizado para numerosos proyectos universitarios y varios trabajos independientes de desarrollo.

Otra de las razones por las cuales se optó por la utilización de Apache, la más importante a mencionar, es la compatibilidad que existe entre las diversas plataformas existentes, de esta manera se podrá realizar la implantación ya sea en un sistema operativo Windows o Unix.

Además al ser Apache uno de los servidores de aplicaciones más utilizados, existe gran cantidad de soporte y ayuda, que facilita la implantación y el manejo de esta herramienta.

4.4.5 *GESTOR DE BASE DE DATOS*

Como sistema gestor de base de datos (DBMS), se escogió MySQL, por la experiencia de haber trabajado algunos proyectos con esta herramienta, teniendo así un mayor dominio en su uso y mejor rendimiento de la misma. El cliente para este gestor de base de datos será PHPMyAdmin.

4.4.6 *LENGUAJE DE PROGRAMACIÓN*

La gran compatibilidad e integración existente con el servidor de aplicaciones y con el gestor de base de datos es una de las causas por las cuales se ha seleccionado a PHP como el lenguaje de programación a utilizarse.

Cabe recalcar que el lenguaje es bastante expansible y posee un gran número de librerías y módulos que mejoran su funcionalidad, permitiendo tener aplicaciones

web mucho más dinámicas y permitiendo la integración con JavaScript, para mejorar considerablemente la operatividad de las aplicaciones.

4.4.7 ENTORNO DE DESARROLLO INTEGRADO (IDE)

Como IDE de desarrollo se decidió por utilizar Notepad++, este editor nos permitirá desarrollar la aplicación en PHP, proporcionando varias ayudas tanto para la generación del código requerido, como para la administración en si del proyecto y la depuración del código generado.

4.4.8 ESTÁNDARES Y BUENAS PRÁCTICAS PARA ENTORNOS WEB

El presente proyecto al estar orientado a un entorno web debe regirse a los estándares web vigentes y que rigen la navegabilidad de la mayoría de las páginas que circulan en Internet. Además, debiendo orientarse a cumplir con la norma ISO/IEC 9216 de calidad de software.

Esto se debe a que cada vez existe mayor diversidad de navegadores, y usuarios que piden más velocidad de navegación, lo que es posible con el uso de patrones, mejorando la accesibilidad y reduciendo los costos que implican el rediseño o el mantenimiento del sitio.

Uno de los principales estándares al cual se hará referencia es el que ha sido publicado por la Organización Internacional para la Estandarización (ISO), bajo la norma ISO/IEC 9216.

4.4.9 DISEÑO DEL CONTENIDO

4.4.9.1 Modelo De Contenido Conceptual

Un modelo conceptual de la interfaz Web de usuario describe los conceptos y las relaciones de la aplicación, y es una base para definir el contenido y la estructura de navegación.

Este modelo puede ser desarrollado mediante el análisis de las tareas, estructuras de información existentes, los roles de usuario o de las expectativas resultantes del uso de otros sistemas similares.

Con esto se debe obtener el contenido que cumpla las siguientes premisas:

- El contenido siempre debe ser adecuado a los efectos de la aplicación web, el público objetivo (caracterizados por diferentes objetivos, preferencias, etc.) y sus tareas.
- El contenido de un sitio web debe ser lo suficientemente completo con respecto a la finalidad deseada y la información que cubra las necesidades del usuario. Los usuarios esperan un sitio web que contenga todos o la mayoría de los contenidos adecuados para su tarea particular y sus necesidades de información.
- Las unidades de contenido deben tener un nivel adecuado detalle, especialmente si esta se la va a utilizar en diferentes partes de la web o se la utilizará para diferentes propósitos.

4.4.9.2 Objetos de Contenido y Funcionalidad

Basado en el modelo de contenido conceptual, los objetos de contenido que se han desarrollado se pueden representar como texto, imágenes, animaciones u otro tipo de medios.

El contenido debe permitir una fácil adaptación de su presentación o estructura a los requisitos cambiantes de los usuarios. Esto puede ser facilitado manteniendo el contenido, su estructura y presentación independientes uno del otro, para mejorar la accesibilidad.

Las técnicas para ello incluyen las hojas de estilo en cascada (CSS), XML (eXtensible Markup Language) y las funciones previstas por un sistema de gestión de contenido.

Se debe tomar en cuenta que:

- Se podrán utilizar objetos de medios de comunicación basados en el tipo de contenido a presentarse y las tareas del usuario, con el propósito de atraer la atención del usuario al sitio o página web sin sobrecargar su percepción.
- Todos los objetos no-textuales, como imágenes o vídeo, deben disponer de descripciones alternativas en texto equivalente y/o con una funcionalidad similar basada en texto.
- Ningún contenido caducado debe mostrarse al usuario, ya que los usuarios esperan que el contenido de un sitio web esté actualizado.
- Debe existir un medio de comunicación entre el usuario con el propietario o desarrollador del sitio Web, así como un mecanismo de retroalimentación.

El sitio web a diseñarse debe adaptar el contenido y la navegación de su interfaz a usuarios individuales o grupos de usuarios, para proveer un acceso más eficiente a la información requerida.

4.4.9.3 Navegación

El diseño de navegación consiste en asignar apropiadamente la estructura del contenido a la estructura navegacional del sitio. Las funciones de búsqueda ofrecen acceso directo a los contenidos, siempre que el contenido sea recuperado en la búsqueda.

La navegación debe ser diseñada para ayudar a los usuarios a comprender dónde están, donde han estado y a dónde pueden ir. Para ello es recomendado que:

- Cada segmento de la presentación (página o ventana) debe proporcionar al usuario una indicación suficiente clara de donde está dentro de la estructura de navegación y de la posición actual del segmento con respecto a la estructura general.

- Deben ofrecerse rutas de acceso alternativas para navegar a una unidad específica de contenidos con el fin de apoyar las diferentes estrategias de navegación.
- El número de pasos de navegación necesarios para alcanzar algún contenido deben ser mínimos para todos los diferentes modelos mentales, las estrategias de navegación y las tareas del usuario.

4.4.9.3.1 Estructura de Navegación

La estructura de navegación determina todos los caminos posibles por los cuales los usuarios pueden desplazarse en la interfaz de usuario Web. Se Deben tomar en cuenta estas recomendaciones:

- Las estructuras de navegación deben ser diseñadas en base a la estructura del modelo de contenido conceptual y estrategias de navegación del usuario, enfocándose a los conceptos relevantes para el mismo.
- Tareas frecuentes pueden ser apoyadas con "enlaces rápidos" a esas tareas en la página principal.
- La estructura de navegación debe ofrecer enlaces identificables a la página relacionada a la tarea que se desea realizar y debe guiar al usuario a través de tareas con múltiples pasos.
- Si se realizara una tarea de múltiples pasos se debe proporcionar:
 - Instrucciones paso a paso.
 - Posición del usuario en la tarea en todo momento.
 - Navegabilidad a pasos anteriores con posibilidad de corregir errores.
 - Retroalimentación sobre el estado del procesamiento de los datos.
- La página de inicio debe proveer suficiente información auto-contenida para que el usuario pueda entender el propósito y anticipar el contenido del sitio, además de proporcionar acceso a las subestructuras de navegación relevantes. Se debe evitar la sobrecarga de sus elementos con un diseño adecuado.
- Evitar páginas innecesarias, tal como páginas pop-up o pantallas splash. Una ventana adicional apoya las tareas del usuario cuando se

utiliza para realizar una subtarea tangencial a la tarea principal en la ventana principal.

4.4.9.3.2 Componentes de Navegación

Los componentes de navegación son grupos de elementos de navegación que permiten a los usuarios moverse en la estructura de navegación conceptual. Los componentes de navegación pueden ayudar a los usuarios a tener una visión general de la estructura de navegación, a tener una idea de a dónde ir o para obtener acceso directo a los nodos relevantes en la estructura (hitos).

Se deben adoptar las siguientes recomendaciones:

- Los enlaces de navegación principal deben ser permanentemente visibles, o deberían volverse fácilmente visibles si se han desplazado fuera de la vista del usuario.
- Si la estructura de navegación tiene varios niveles, el componente de navegación debe ser diseñado para mostrar más de un nivel de navegación al mismo tiempo.
- Una vista general de la navegación, tal como un mapa del sitio, debe proveerse mostrando la estructura del sitio en una forma resumida.
- Evitar enlaces muertos y enlaces incorrectos.

4.4.9.4 Presentación del Contenido

Se recomiendan aspectos de presentación del contenido que se pueden especificar por separado, utilizando por ejemplo hojas de estilo.

- Si las páginas son diferentes y tienen diferentes propósitos y tipos de contenido, pueden agruparse en diferentes categorías, utilizando un esquema de diseño para cada categoría de forma consistente.
- Los títulos de página deben colocarse en una posición coherente en todas las páginas.
- Las páginas cortas son más apropiadas para páginas de inicio, páginas de navegación, o páginas de resumen, que hay que leerlas con rapidez. Las páginas largas pueden ser más apropiadas cuando los usuarios desean leer el contenido sin interrupciones.

- El desplazamiento (scroll) horizontal se debe evitar siempre que sea posible.
- El color debe utilizarse con cuidado, teniendo en cuenta que no es el único medio para transmitir información, prestando atención a:
 - Limitar el número de colores utilizados para propósitos de codificación a no más de diez, y preferiblemente no más de cinco.
 - Mantener contraste entre el fondo de la página con el primer plano, evitando combinaciones de colores que puedan impedir la lectura del texto.
 - Todas las páginas o ventanas que pertenecen a un sitio web deben ser fácilmente identificables como parte de ese sitio.
 - Los enlaces deben ser fácilmente reconocibles por el usuario con el uso de técnicas como el subrayado y el código de colores de texto para resaltarlos de un grupo de elementos de navegación. Es importante que los enlaces no sean reconocibles sólo por su color.
 - Generalmente la mejor técnica para transmitir el significado de un enlace para el usuario es por uso de símbolos gráficos que representen algún objeto relacionado para el efecto. Por ejemplo, un símbolo de coche de compras en una aplicación de comercio electrónico.
 - Los nombres de los enlaces textuales deben ser suficientemente largos para ser entendido, pero suficientemente cortos como para evitar confusión.
 - Los enlaces que abren nuevas ventanas del navegador o ventanas pop-up deben estar claramente marcados, por un símbolo gráfico especial, como una pequeña flecha que se utiliza sistemáticamente para indicar ese efecto.
 - La lectura y la comprensión del contenido textual deben ser apoyadas por el uso de frases cortas, división del texto en fragmentos cortos o por el uso de viñetas, sin descuidar la ortografía y la gramática.

4.4.9.5 Aspectos Generales de Diseño

El contenido o la funcionalidad pueden no ser obvios para todos los usuarios, por lo que debe proporcionarse información de ayuda adecuada con enlaces claramente identificables que conducen a páginas de ayuda.

Deben minimizarse los errores potenciales de los usuarios, así como el esfuerzo para recuperarse de ellos.

El contenido de los mensajes de error que aparezcan en las páginas deben mostrar claramente la razón por la cual se produjo el error y, si es posible, las acciones que el usuario puede tomar para resolver el error.

El uso de estándares tales como XHTML, CSS y otros, reduce el riesgo de presentar la interfaz de usuario Web de manera inadecuada. De lo contrario, la usabilidad y la accesibilidad de la interfaz de usuario web puede verse afectada negativamente.

Se deben trabajar con diferentes tecnologías de uso común (como diferentes navegadores) o características técnicas típicas (como tamaños de pantalla).

Además, para asegurar la calidad del sitio web, estos requisitos deben intentar ser cumplidos en su totalidad.

4.5 ANÁLISIS Y DISEÑO DEL SISTEMA

4.5.1 *ESPECIFICACIÓN DE REQUERIMIENTOS*

Los requerimientos que inicialmente se obtuvieron de citas con los representantes del Colegio Sultana del Oriente fueron traducidos a Historias de Usuario, y se las agruparon por módulos según el servicio que se vaya a prestar a los usuarios del Sistema.

4.5.2 *USUARIOS DEL SISTEMA*

Los usuarios que han sido contemplados para la planificación global del sistema cumplen un rol fundamental en cada una de las Historias de Usuario a describirse a continuación, ya que son los que obtendrán un beneficio real por el uso apropiado del Sistema a desarrollarse.

Administrador: Este usuario tiene la función de configurar todos los parámetros del sistema para realizar la gestión académica, tal como gestionar materias, usuarios, evaluaciones, reportes, ver notas, tareas y notificaciones, además del ingreso de nuevos usuarios, alumnos y de nuevos profesores. Es responsable de la planificación de los periodos lectivos y la generación de certificados de matrícula.

Autoridad/Secretaría: Este usuario tiene la posibilidad de ingresar y editar información de docentes, estudiantes, dar reportes de notas a los estudiantes, realizar y editar matriculas, revisar su agenda telefónica, obtener reportes de docentes, estudiantes y cursos.

Alumno: Estos usuarios tienen la posibilidad de realizar consultas específicas para el alumno que ha sido ingresado al sistema, tal como las notas, cambios de algunos datos de su perfil, su clave de usuario, ver notificaciones, tareas y su agenda telefónica, recalando que el estudiante deberá obligatoriamente realizar las evaluaciones asignadas para poder entrar a su plataforma.

Profesor: Este usuario tiene la función de ingresar las notas de los diferentes aportes para todos y cada uno de los alumnos de las materias en las cuales el profesor fue planificado en los quimestres que posee cada año lectivo, además de actualizar algunos datos de su perfil, cambiar su clave de usuario, ver su agenda telefónica consultar notas, crear tareas y notificaciones, e ingresar al módulo de evaluación curricular para ingresar las unidades y temas a él designadas.

4.5.2.1 HISTORIAS DE USUARIO

En la elaboración de las Historias de Usuario es necesario tomar en cuenta parámetros básicos asociados al desarrollo, tales como riesgo y esfuerzo.

Como riesgos primordiales del sistema se pueden mencionar el manejo de tecnología no conocida o probada y las restricciones de disponibilidad de recursos humanos, de hardware o de software.

El esfuerzo es calculado en base a puntos, que corresponden a semanas ideales de trabajo, tomando en cuenta el esfuerzo asociado con pruebas unitarias, integración de módulos, pruebas de aceptación y refactorización del código.¹⁰

Cada Historia de Usuario debe contener los siguientes elementos:

- Número:
Identificador numérico de la historia de usuario.
- Nombre:
Denominación que se le da a la historia de usuario para facilitar su posterior identificación.
- Usuario:
Usuario dentro del sistema que realizará la operación detallada en la Historia de Usuario.
- Riesgo en desarrollo:
Se lo define basado en el riesgo que afronta el equipo de desarrollo en la obtención de resultados de una historia de usuario, para que satisfaga los requerimientos del cliente.

Puede ser alto, medio o bajo.
- Prioridad en Negocio:
Definido por el cliente, es el grado de prioridad para el desarrollo de cada historia de usuario.
- Puntos Estimados:
Puntos de esfuerzo determinado por la duración de desarrollo de la historia de usuario.
- Descripción:
Explicación del requerimiento expresado en la historia de usuario. Puede ser susceptible a cambios durante la etapa de desarrollo.

¹⁰ <https://pid.dsic.upv.es/CI/Material/Documentos%20Disponibles/Caso de Estudio XP.doc>

- Detalles:

Opcional. Breves puntos que describan o aclaren la descripción y la historia de usuario en sí.

4.5.2.1.1 Módulo de Parametrización del Sistema

HISTORIA DE USUARIO	
Número: 01	Nombre: Ingreso de estudiantes y profesores
Usuario: Administrador	Riesgo de desarrollo: Medio
Prioridad en Negocio: Alta	Puntos Estimados: 0,4
Descripción: El Administrador debe ingresar los datos personales de los estudiantes y los profesores, para posterior asignación de cursos y distribución de paralelos. Estos datos pueden ser modificados o eliminados según el Administrador lo necesite.	
Observaciones: Los datos personales constan de nombres y apellidos, número de cédula de identidad, fecha de nacimiento, teléfonos, correo electrónico, lugar de nacimiento y dirección domiciliaria.	

Tabla 4 Historia de Usuario -: Ingreso de estudiantes y profesores

HISTORIA DE USUARIO	
Número: 02	Nombre: Creación de Usuarios en el Sistema
Usuario: Administrador	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Alta	Puntos Estimados: 0,2
Descripción: El Administrador debe tener la capacidad de crear, modificar y eliminar usuarios para el sistema, así como asignarle un perfil dentro del mismo.	
Observaciones:	

Tabla 5 Historia de Usuario:-Creación de usuarios en el Sistema.

HISTORIA DE USUARIO	
Número: 03	Nombre: Ingreso de Cursos
Usuario: Administrador	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Alta	Puntos Estimados: 0,2
Descripción: El Administrador debe tener la capacidad de crear, modificar y eliminar cursos en el sistema.	
Observaciones: Los cursos se crean antes de paralelos y especialidades.	

Tabla 6 Historia de Usuario:-Ingreso de Cursos.

4.5.2.2 Módulo de Planificación Académica

HISTORIA DE USUARIO	
Número: 04	Nombre: Matriculación
Usuario: Administrador	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Alta	Puntos Estimados: 1
Descripción: El Administrador debe matricular por cada periodo lectivo a todos los alumnos del colegio, de acuerdo a la planificación previamente realizada con los horarios, materias y profesores.	
Observaciones: Para poder matricular a los alumnos en un cierto periodo lectivo, se debe realizar la respectiva planificación considerando los horarios en los cuales se dictará cada materia, con la asignación respectiva del profesor para la misma.	

Tabla 7 Historia de Usuario:-Matriculación.

HISTORIA DE USUARIO	
Número: 05	Nombre: Asignación de Docentes
Usuario: Administrador	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Alta	Puntos Estimados: 1

<p>Detalle:</p> <p>El Administrador debe asignar a cada uno de los docentes a un curso y un paralelo por cada año lectivo, de esto se determinará las materias que deberá dictar en el periodo asignado.</p>
<p>Observaciones:</p> <p>Debe estar creados los cursos y paralelos en los cuales se les asignará a los docentes, además la planificación de las materias por cada uno de estos.</p>

Tabla 8 Historia de Usuario:-Asignación de Docentes

HISTORIA DE USUARIO	
Número: 06	Nombre: Registro de Notas por Alumno
Usuario: Docente	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Alta	Puntos Estimados: 0,3
<p>Detalle:</p> <p>El profesor deberá ingresar las notas de los alumnos por cada uno de los aportes del bimestre, además de una observación a dichas notas al final del año lectivo.El sistema deberá realizar los promedios respectivo que determinen si el alumno queda suspenso, aprueba o pierde la materia.</p>	
<p>Observaciones:</p> <p>Previo al registro de notas cada alumno debió ser matriculado en el sistema en todas las materias que toma en el periodo lectivo Se debió realizar con anterioridad la planificación de los profesores de acuerdo a las materias y en los horarios a impartirse las materias.</p>	

Tabla 9 Historia de Usuario:-registro de Notas por Alumno

HISTORIA DE USUARIO	
Número: 07	Nombre: Ingreso de Unidades y Temas
Usuario: Docente	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Alta	Puntos Estimados: 0,3
<p>Detalle:</p> <p>El profesor deberá ingresar las unidades y temas de cada unidad en el sistema.</p>	
<p>Observaciones:</p> <p>Previo al ingreso de unidades (Plan Curricular) el docente deberá estar registrado en el sistema, además debe haber cursos, paralelos y especialidades y materias.</p>	

Tabla 10 Historia de usuario:-Ingreso de unidades y temas

HISTORIA DE USUARIO	
Número: 08	Nombre: Ingreso de Materias
Usuario: Administrador	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Alta	Puntos Estimados: 0,3
Detalle: El administrador ingresa las materias correspondientes a cada curso y especialidad.	
Observaciones: Para poder ingresar las materias deben estar creados los cursos y especialidades.	

Tabla 11 Historia de Usuario:-Ingreso de Materias

HISTORIA DE USUARIO	
Número: 09	Nombre: Ingreso de Especialidades
Usuario: Administrador	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Alta	Puntos Estimados: 0,3
Detalle: El administrador ingresa las especialidades de la institución.	
Observaciones:	

Tabla 12 Historia de Usuario:-Ingreso de Especialidades.

HISTORIA DE USUARIO	
Número: 10	Nombre: Ingreso de Paralelos
Usuario: Administrador	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Alta	Puntos Estimados: 0,3
Detalle: El administrador ingresa los paralelos para cada curso y especialidad.	
Observaciones:	

Tabla 13 Historia de Usuario:-Ingreso de Paralelos

4.5.2.3 Módulo de Evaluación.

HISTORIA DE USUARIO	
Número: 11	Nombre: Ingreso de Unidades y Temas
Usuario: Administrador	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Media	Puntos Estimados: 0,2
Detalle: El administrador ingresa las unidades y temas de las materias a ser evaluadas.	
Observaciones: Deben estar creados los cursos paralelos especialidad y docente. Las unidades tendrán descripción y objetivo.	

Tabla 14 Historia de Usuario:-Ingreso de Unidades y Temas

HISTORIA DE USUARIO	
Número: 12	Nombre: Ingreso de Unidad a ser Evaluada
Usuario: Administrador	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Media	Puntos Estimados: 0,2
Detalle: El administrador ingresa la unidad a ser evaluada por los alumnos.	
Observaciones: Deben estar ingresadas las unidades y temas de todas las materias del curso, además una unidad no debe estar sin temas. El administrador es responsable de ingresar unidades consecutivas ascendentes.	

Tabla 15 Historia de Usuario:-Ingreso de Unidad a ser Evaluada

HISTORIA DE USUARIO	
Número: 13	Nombre: Evaluación de Unidades por Alumno
Usuario: Alumno	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Alta	Puntos Estimados: 0,3

Detalle: El estudiante evaluara los temas de cada unidad de cada materia del curso tomado.
Observaciones: El estudiante para poder entrar a su plataforma en el sistema debe obligatoriamente evaluar la unidad asignada por el administrador. Para cada evaluación el profesor de cada materia tiene la obligación de enviar por lo menos una tarea por medio del sistema (internet).

Tabla 16 Historia de Usuario:-Evaluación de Unidades por Alumno

4.5.2.4 Módulo de Reportes.

HISTORIA DE USUARIO	
Número: 14	Nombre: Reporte de Listado de Cursos
Usuario: Secretaria	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Media	Puntos Estimados: 0,3
Detalle: El usuario podrá sacar reportes de listas de cursos por curso y especialidad.	
Observaciones:	

Tabla 17 Historia de Usuario:-Reporte de Listado de Cursos

HISTORIA DE USUARIO	
Número: 15	Nombre: Nómina General de Estudiantes
Usuario: Secretaria	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Media	Puntos Estimados: 0,2
Detalle: El usuario podrá obtener reportes de todos los alumnos del plantel ordenados por especialidad.	
Observaciones:	

Tabla 18 Nómina General de Estudiantes

HISTORIA DE USUARIO	
Número: 16	Nombre: Nómina General de Docentes
Usuario: Secretaria	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Media	Puntos Estimados: 0,2
Detalle: El usuario podrá obtener reportes de todos los docentes del plante.	
Observaciones:	

Tabla 19 Historia de Usuario:-Nómina General de Docentes

HISTORIA DE USUARIO	
Número: 17	Nombre: Reporte de Resultados de Evaluación Curricular
Usuario: Administrador	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Alta	Puntos Estimados: 0,4
Detalle: El Administrador podrá obtener resultados de la evaluación curricular filtrado por curso, especialidad, materia y docente.	
Observaciones: Se generara el reporte con el número de alumnos que hayan evaluado hasta ese momento. Se presentara el porcentaje cumplido de cada tema de cada unidad.	

Tabla 20 Historia de Usuario:-Reporte de Resultados de Evaluación Curricular

4.5.2.5 Módulo de comunicación.

HISTORIA DE USUARIO	
Número: 18	Nombre: Envío de tareas a los alumnos
Usuario: Docente	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Media	Puntos Estimados: 0,2
Detalle: El docente puede enviar tareas por medio de la plataforma informática a sus alumnos.	

<p>Observaciones:</p> <p>Los alumnos estarán en la obligación de revisar su plataforma de acuerdo a las sugerencias del docente.</p>

Tabla 21 Historia de Usuario:-Envío de Tareas a los Alumnos

HISTORIA DE USUARIO	
Número: 19	Nombre: Envío de notificaciones a los alumnos
Usuario: Docente	Riesgo de desarrollo: Bajo
Prioridad en Negocio: Media	Puntos Estimados: 0,2
<p>Detalle:</p> <p>El docente puede enviar notificaciones por medio de la plataforma informática a sus alumnos de forma individual.</p>	
Observaciones:	

Tabla 22 Historia de Usuario:-Envío de Notificaciones a los alumnos

4.6 ANÁLISIS DEL SISTEMA

Una vez definidas las funcionalidades que debe contener el sistema, es primordial realizar un análisis de tiempo y esfuerzo que va a ser requerido en el transcurso del proyecto, esto para obtener una estimación real de recursos que el equipo de desarrollo demanda para cumplir con los requerimientos planteados en las Historias de Usuario.

Para ello la interacción entre representantes del Colegio Sultana del Oriente y el equipo de desarrollo es imprescindible como retroalimentación dentro del proceso mismo del perfeccionamiento del sistema.

4.6.1 ESTIMACIÓN DE ESFUERZOS

Para realizar una estimación certera y lo más cercana al tiempo real de desarrollo, se ha considerado los siguientes ámbitos:

- Cada punto de esfuerzo es tomado como cinco días laborables.

- Se asume un calendario por tiempo¹¹.
- Los horarios de trabajo son de 8 horas.
- Se asigna un equipo de desarrollo de dos personas.

No. Historia	Nombre de Historia	Riesgo	Prioridad	Puntos Estimados	Equivalencia en tiempo (días)
01	Ingreso de estudiantes y profesores	Medio	Alta	1	5
02	Creación de Usuarios en el Sistema	Medio	Alta	0,8	4
03	Ingreso de Cursos	Bajo	Alta	0,6	3
04	Matriculación	Medio	Alta	1,6	8
05	Asignación de Docentes	Medio	Alta	1	5
06	Registro de Notas de Alumnos	Bajo	Alta	1,2	6
07	Ingreso de Unidades y Temas	Medio	Alta	1	5
08	Ingreso de Materias	Bajo	Alta	0,2	1
09	Ingreso de Especialidades	Bajo	Alta	0,2	1
10	Ingreso de paralelos	Bajo	Media	0,2	1
11	Ingreso de unidades y Temas	Medio	Alta	1	5
12	Ingreso de Unidad a ser Evaluada	Bajo	Alta	0,2	1
13	Evaluación de Unidades por Alumno	Medio	Alta	0,6	3
14	Reporte de Listado de Cursos	Medio	Baja	0,2	1
15	Nómina General de Estudiantes	Bajo	Baja	0,2	1
16	Nómina General de Docentes	Bajo	Baja	0,2	1
17	Reporte de Resultado de Evaluación Curricular	Medio	Alta	1,6	8
18	Envío de tareas a los Alumnos	Bajo	Media	0,4	2
19	Envío de Notificaciones a los Alumnos	Bajo	Baja	0,4	2

Tabla 23 Estimación de Esfuerzos

4.6.2 PRIORIZACIÓN

De la estimación de esfuerzo realizada, tomando en cuenta su costo en tiempo, la prioridad que representa para el Colegio Sultana del Oriente y el riesgo que pueda presentarse por diversos factores en el desarrollo, se puede establecer el orden en el cual las historias de usuario van a ser entregadas, designándoles una iteración tentativa.

¹¹ BECK, Kent; FOWLER, Martin. "Planning Extreme Programming"

No. Historia	Nombre de Historia	Iteración
01	Ingreso de estudiantes y profesores	1
09	Ingreso de Especialidades	1
03	Ingreso de Cursos	1
11	Ingreso de paralelos	1
08	Ingreso de Materias	1
02	Creación de Usuarios en el Sistema	1
07	Ingreso de Unidades y Temas (Docentes)	1
05	Asignación de Docentes	2
04	Matriculación	2
06	Registro de Notas de Alumnos	2
12	Ingreso de Unidades y Temas (Administrador)	2
13	Ingreso de unidad a ser evaluada	2
13	Evaluación de Unidades por Alumno	3
14	Reporte de Listado de Cursos	3
15	Nómina General de Estudiantes	3
16	Nómina General de Docentes	3
17	Reporte de Resultado de Evaluación Curricular	3
18	Envío de tareas a los Alumnos	3
19	Envío de Notificaciones a los Alumnos	3

Tabla 24 Priorización

4.6.4 PLAN DE ENTREGAS

Dentro de cada iteración ya se puede establecer fechas reales de desarrollo para cada historia de usuario descrita previamente. Algunas de estas historias, con sus respectivas tareas subyacentes, pueden ser realizadas en paralelo por lo que las fechas en las que serán desarrolladas coincidirán. Esto disminuye considerablemente el tiempo total del proyecto.

No. Historia	Nombre de Historia	Iteración	Fecha Inicio	Fecha Fin
01	Ingreso de estudiantes y profesores	1	2/04/2012	7/04/2012
09	Ingreso de Especialidades	1	10/04/2012	11/04/2012
03	Ingreso de Cursos	1	12/04/2012	16/04/2012
10	Ingreso de paralelos	1	17/04/2012	18/04/2012
08	Ingreso de Materias	1	19/04/2012	20/04/2012
02	Creación de Usuarios en el Sistema	1	23/04/2012	26/04/2012
07	Ingreso de Unidades y Temas (Docentes)	1	27/04/2012	03/05/2012
05	Asignación de Docentes	2	4/05/2012	08/05/2012
04	Matriculación	2	11/05/2012	20/05/2012
06	Registro de Notas de Alumnos	2	21/05/2012	28/05/2012
11	Ingreso de Unidades y Temas (Administrador)	2	29/05/2012	02/06/2012
12	Ingreso de unidad a ser evaluada	2	03/06/2012	05/06/2012
13	Evaluación de Unidades por Alumno	3	30/07/2012	05/08/2012
14	Reporte de Listado de Cursos	3	06/08/2012	07/08/2012
15	Nómina General de Estudiantes	3	08/06/2012	09/08/2012
16	Nómina General de Docentes	3	13/08/2012	14/08/2012

17	Reporte de Resultado de Evaluación Curricular	3	15/08/2012	24/08/2012
18	Envío de tareas a los Alumnos	3	27/08/2012	28/08/2012
19	Envío de Notificaciones a los Alumnos	3	29/08/2012	31/08/2012

Tabla 25 Plan de Entregas

Con esto se puede sugerir un calendario de eventos para el proyecto, el que concluye con la entrega de las funcionalidades planteadas al inicio del desarrollo del sistema.

Evento	Fecha
Inicio del proyecto	2/04/2012
Primera Entrega Realizada	03/05/2012
Segunda Entrega Realizada	05/06/2012
Tercera Entrega Realizada	31/08/2012

Tabla 26 Calendario de Entregas

4.7 DISEÑO DEL SISTEMA

4.7.1 DISEÑO ARQUITECTÓNICO DEL SISTEMA

El diseño arquitectónico en el que se basa el presente sistema se basa en una estructura en tres capas, con sus tres componentes principales¹²:

- Capa de presentación, también denominada capa de usuario, la que presenta el sistema desarrollado vía navegadores web.
- Capa de negocio, que se comunica con la capa de presentación para recibir las solicitudes y presentar resultados, y con la capa de datos para solicitar al Sistema Gestor de la Base de Datos en el almacenamiento o recuperación de los datos.

¹² http://es.wikipedia.org/wiki/Arquitectura_de_tres_niveles

- Capa de datos, donde residen los datos y se accede a ellos. Realiza todo el almacenamiento de datos, recibe las solicitudes de almacenamiento y recuperación de información desde la capa de negocio.

Esto se aprecia claramente en la Ilustración 3.

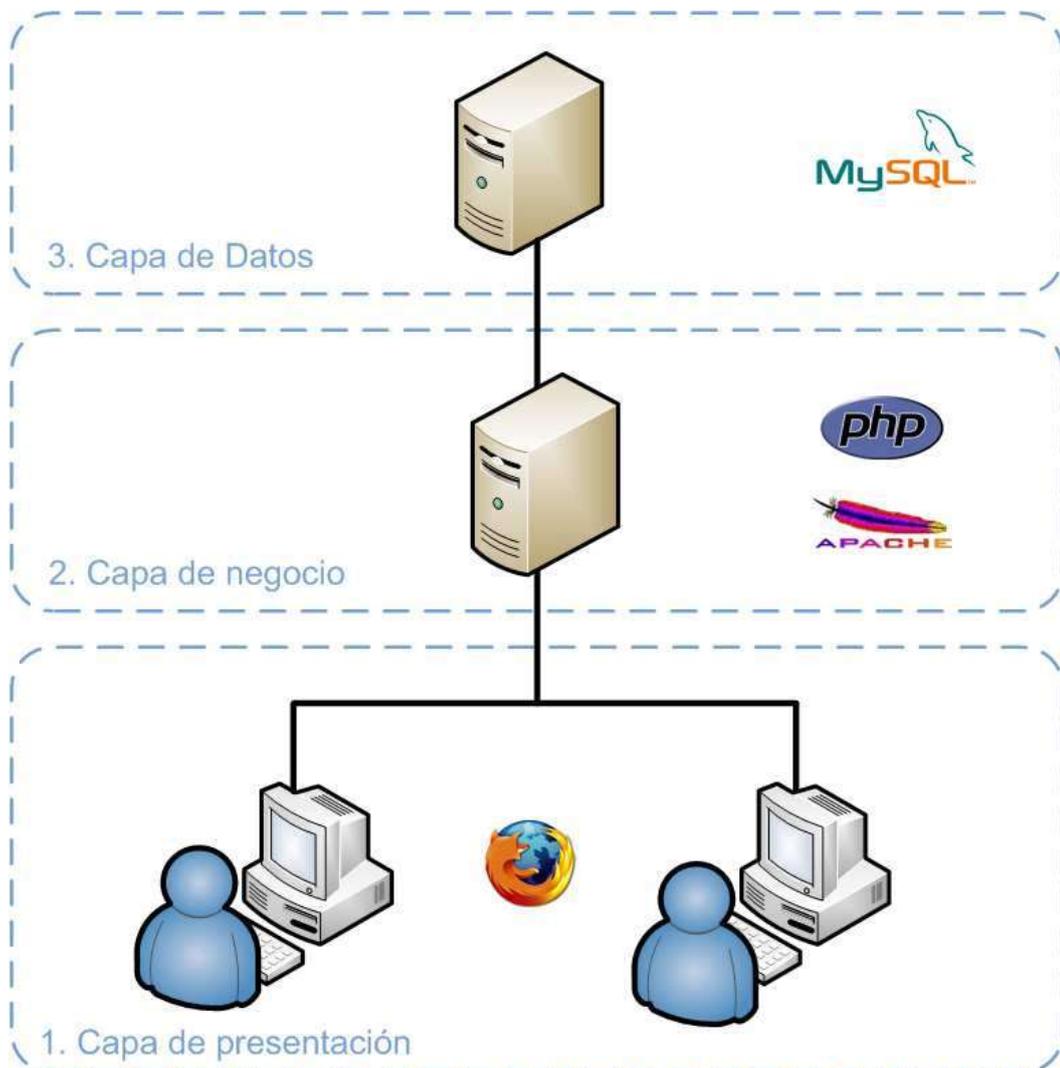


Ilustración 3.-Diseño arquitectónico en tres capas

4.7.2 DISEÑO DEL DIAGRAMA DE CLASES

El siguiente diagrama representa las entidades requeridas para la satisfacción de las necesidades del cliente y las relaciones entre ellas sobre la evaluación curricular.

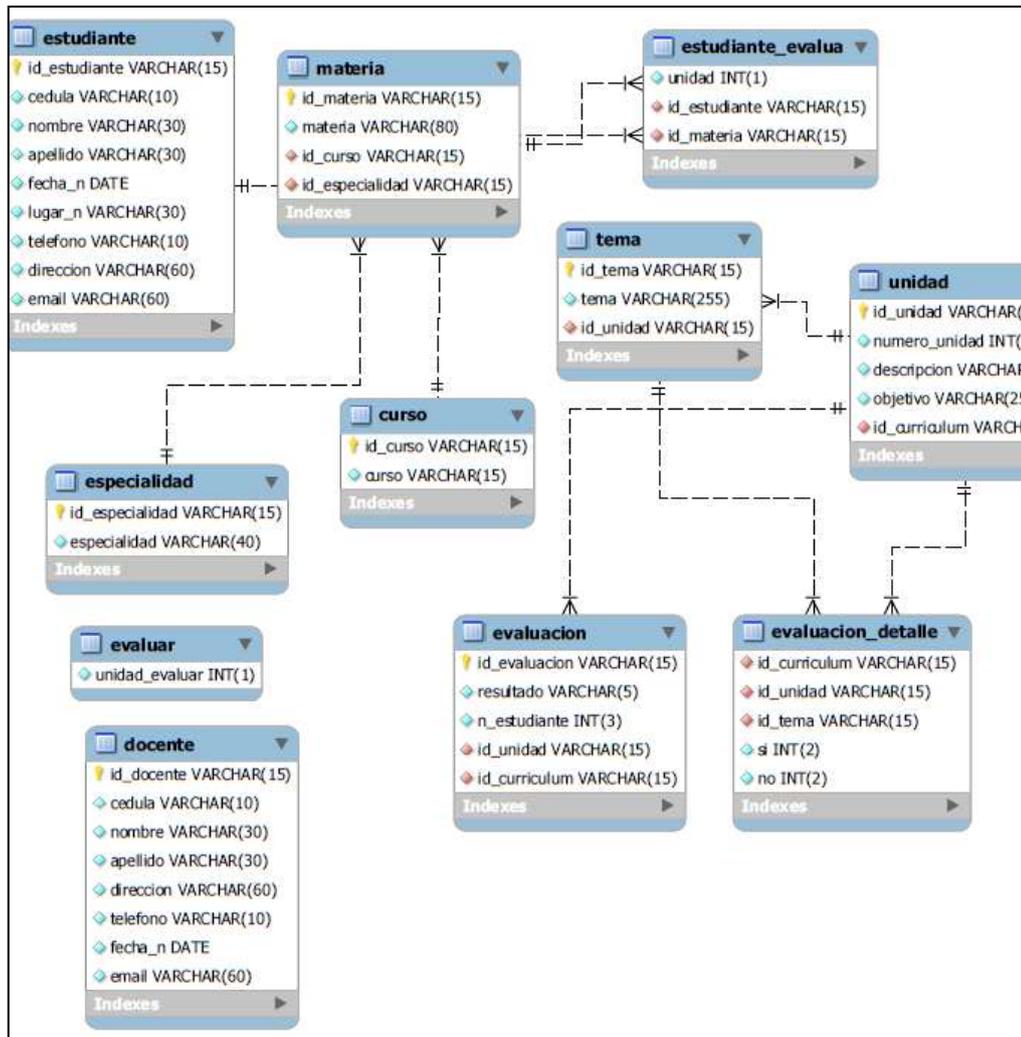


Ilustración 4 Diagrama de clases- Módulo de evaluación

4.7.3 DISEÑO DE DIAGRAMAS DE INTERACCIÓN¹³

A continuación se presenta parte de los diagramas de interacción, más información en el Anexo B

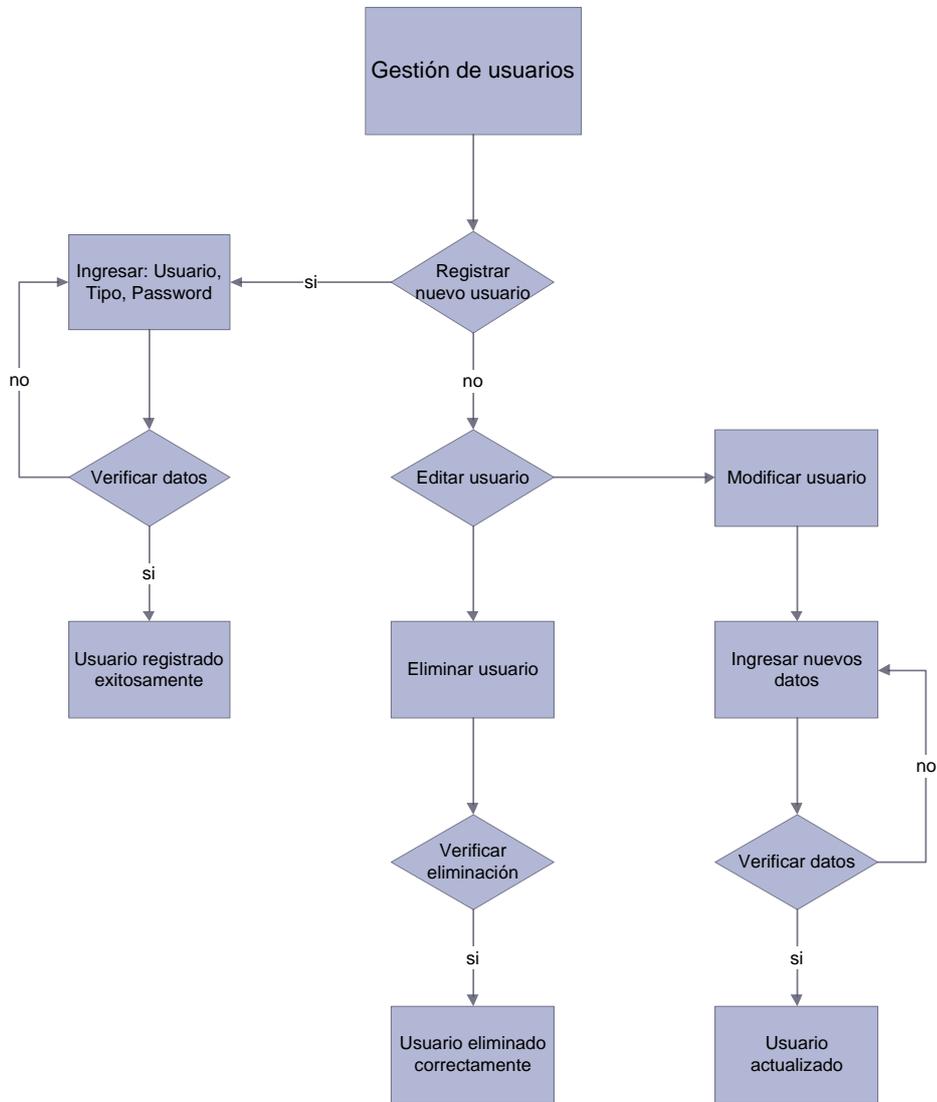


Ilustración 5 Diagrama de Interacción-Gestión de Usuarios

¹³ <http://www.guiaweb.gob.cl/guia/capitulos/tres/experiencia.htm>

4.7.4 GESTIÓN DE MATERIAS A EVALUAR

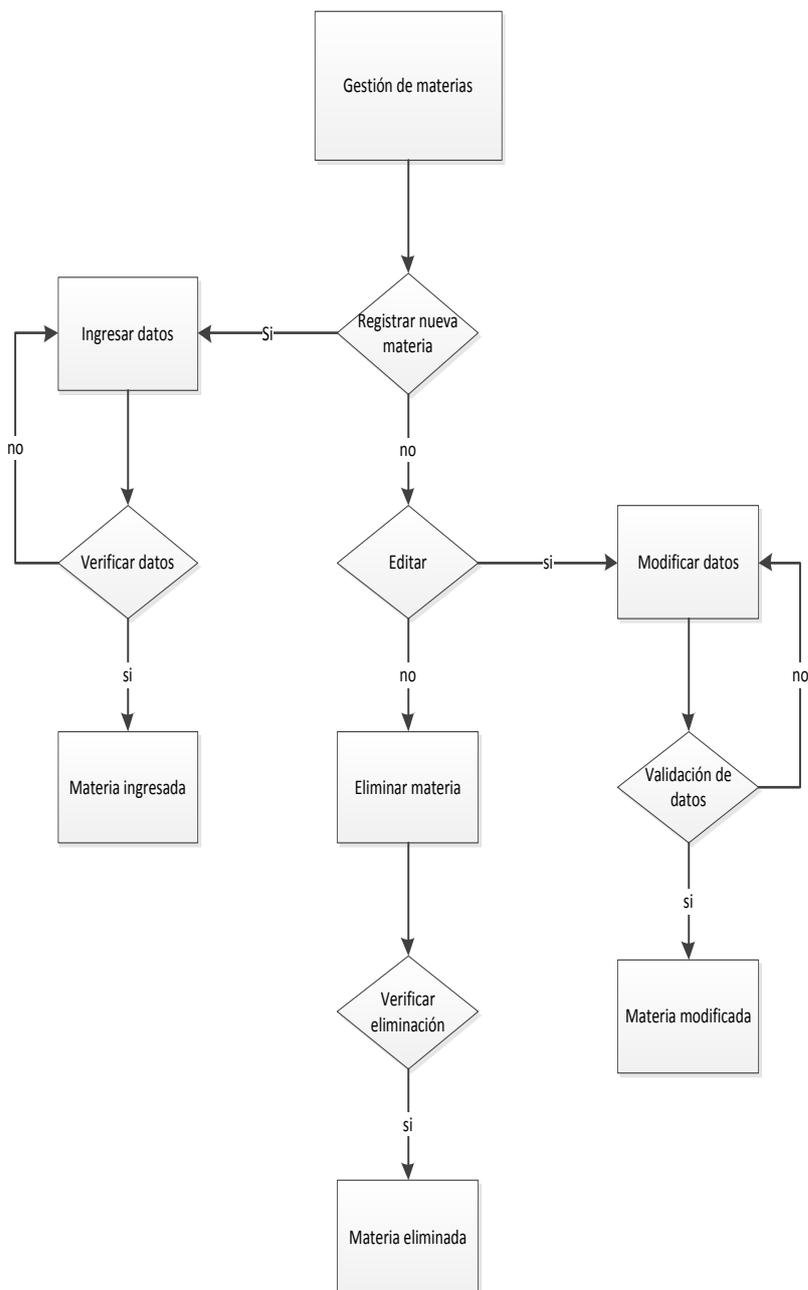


Ilustración 6 Diagrama de Interacción-Ingreso de materias

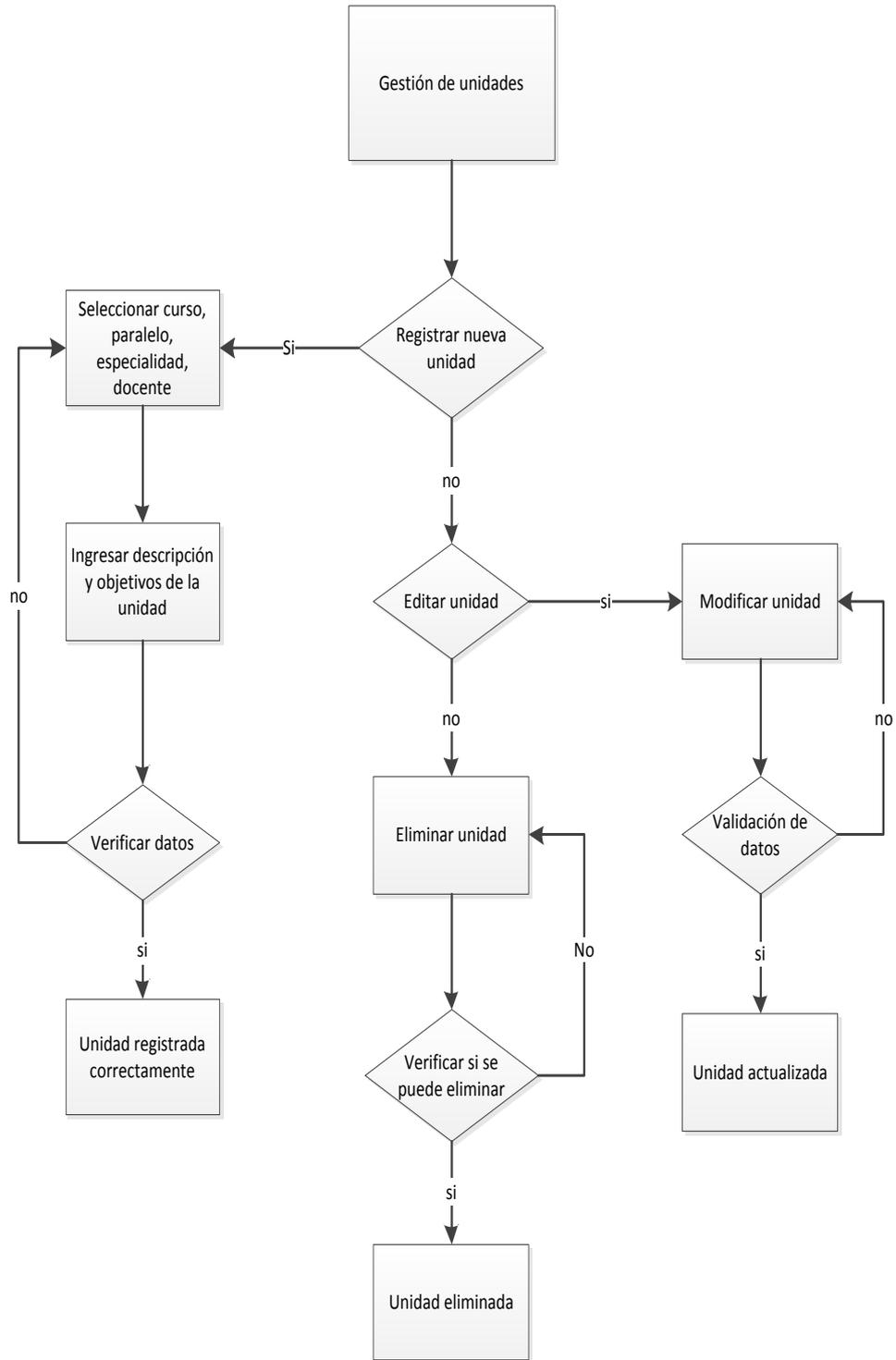


Ilustración 7 Gestión de Evaluación-Ingreso de unidades

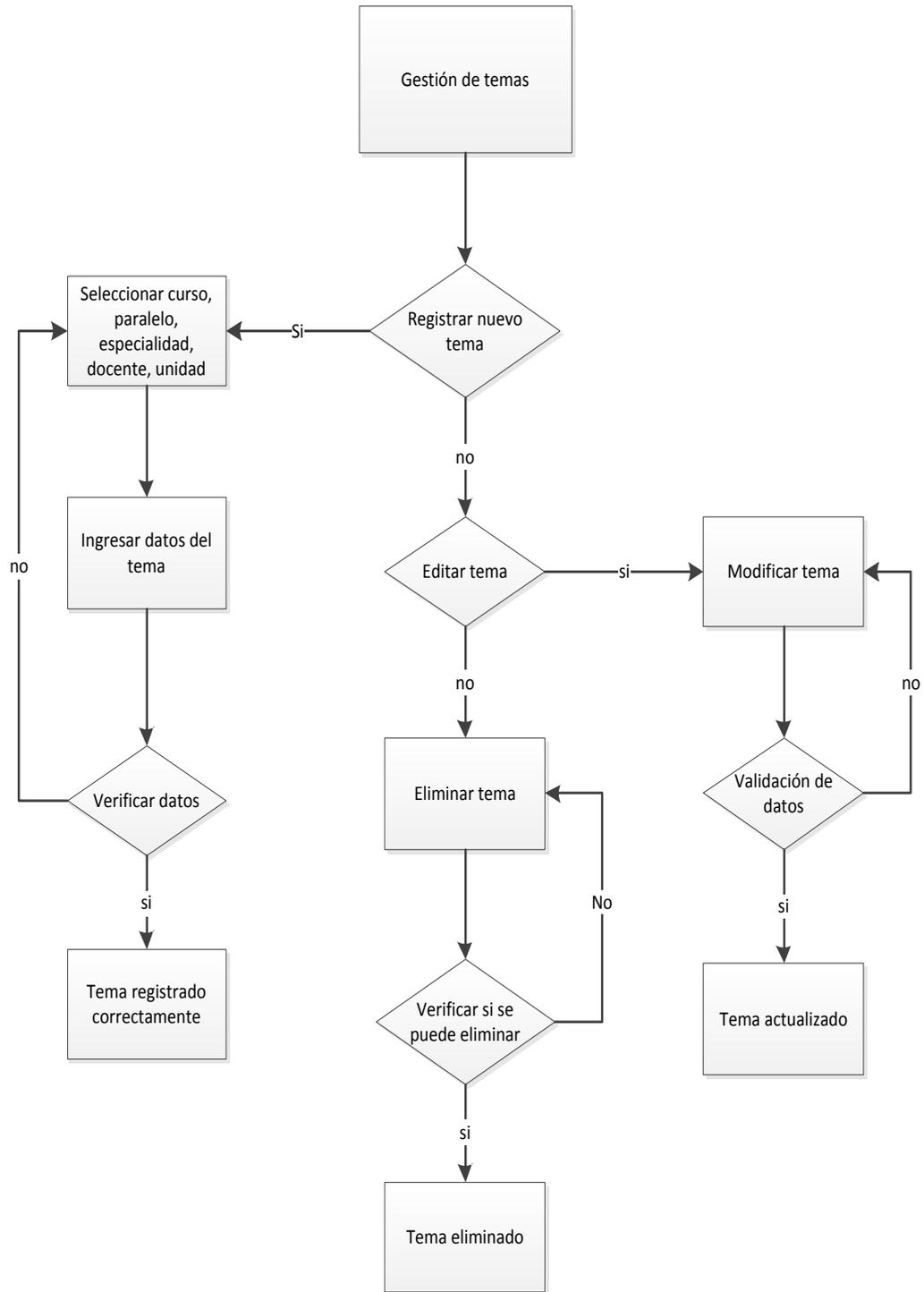


Ilustración 8 Gestión de Evaluación-Ingreso de temas

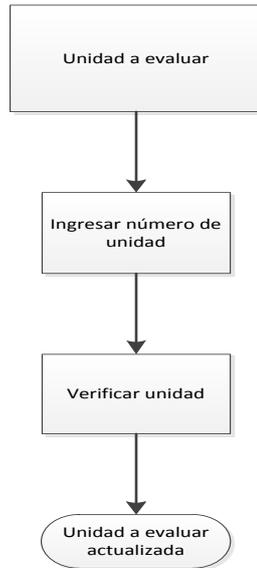


Ilustración 9 Gestión de Evaluación-Ingreso de Unidad a Evaluar

4.7.5 GESTIÓN DE REPORTE DE RESULTADOS



Ilustración 10 Reportes-Reporte Parcial de Resultados

4.7.6 DISEÑO DEL ESQUEMA JERÁRQUICO

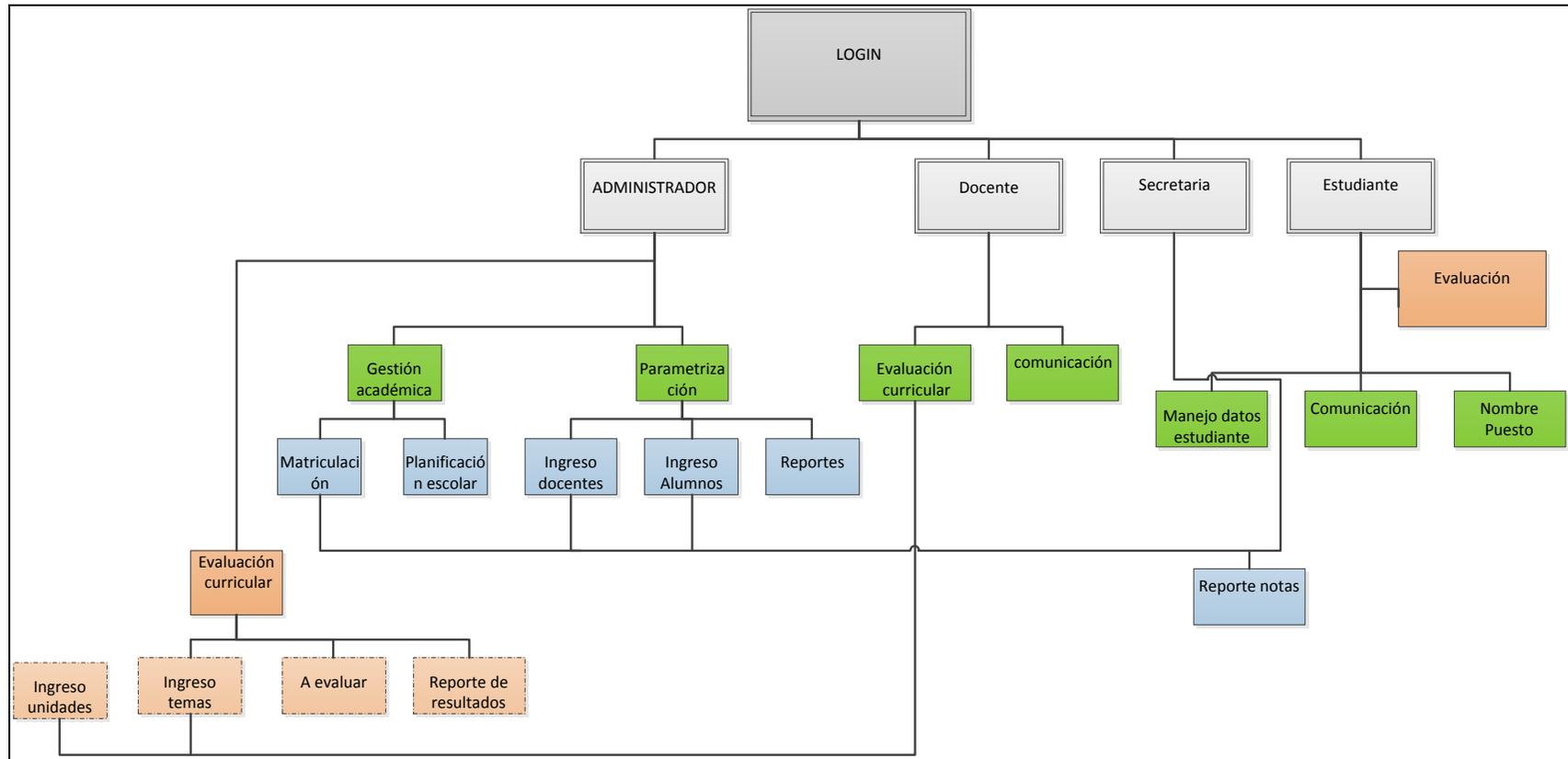


Ilustración 11 Diagrama Jerárquico

4.7.7 DISEÑO DE LAS INTERFACES DE USUARIO

Las interfaces de usuario se rigen a los estándares descritos en el Capítulo Dos, y son presentadas en su totalidad en el **Anexo A**. Las interfaces que se muestran a continuación son muestras de interfaces usadas como plantillas en el proceso de desarrollo.

The screenshot shows a registration form titled "REGISTRO DE USUARIOS EN EL SISTEMA". It contains the following fields: "Usuario:" with a text input, "Tipo:" with a dropdown menu showing "Administrador", "Password:" with a text input, and "Confirmar Pass:" with a text input. A "Guardar" button is located at the bottom center of the form.

Ilustración 12 Gestión de Usuarios

The screenshot shows a registration form titled "REGISTRO DE DOCENTES EN EL SISTEMA". It contains the following fields: "Cédula:" with a text input, "Nombre:" with a text input, "Apellido:" with a text input, "Dirección:" with a text input, "Telefono:" with a text input, "F.Nac.:(aaaa-mm-dd)" with a date input, and "E-mail:" with a text input. A "Guardar" button is located at the bottom center of the form.

Ilustración 13 Gestión de Docentes

The screenshot shows a registration form titled "REGISTRO DE ESTUDIANTES EN EL SISTEMA" with fields for "Cédula:", "Apellido:", "Nombre:", "F.Nac.:(aaaa-mm-dd)", "Dirección:", and "Telefono:". Below the form is a table with the following data:

ID	CÓDULA	NOMBRE	APELLIDO	FECHA NAC.	DIRECCIÓN	TELÉFONO	E-MAIL	ACCIONES
9270-4301261130	90040967	RODRIG	PERANDEZ	0000-00-00	94000	012406331	mrodr@unimab.com	[Iconos]
9270-4301262147	541304504	FRANZ	PEREZ	1970-02-01	94000	221241416	mrodr@unimab.com	[Iconos]
9270-4302460370	540000042	DAVID	REYES	1970-02-03	94000	494238035	mrodr@unimab.com	[Iconos]

Ilustración 14 Gestión de Estudiantes



Ilustración 15 Gestión de Materias



Ilustración 16 Gestión de Evaluación Curricular

4.8 IMPLANTACIÓN, PRUEBAS Y ANÁLISIS DE RESULTADOS

4.8.1 *IMPLANTACIÓN DEL SISTEMA*

4.8.1.1 *ALOJAMIENTO DENTRO DEL WEB HOSTING*

El Sistema de Evaluación Curricular vía Web para el Colegio Sultana del Oriente, fue implantando dentro del Web hosting previamente adquirido por el CSO, mediante el cliente FTP Dreamweaver.

La configuración de la base de datos requerida para el funcionamiento del sistema fue realizada mediante la herramienta PHPMyAdmin, instalada previamente en el hosting de la institución, herramienta con la cual se realiza la administración de la base de datos realizada en MySQL.

4.8.1.2 Experiencia del Usuario

Una vez que el sistema ha sido implantado dentro del hosting del CSO, se realizaron las pruebas respectivas que se detallan en el capítulo “Pruebas del Sistema” para lo cual el usuario final tuvo una previa capacitación de acuerdo a cada uno de los perfiles que se manejan en la aplicación.

La experiencia del usuario fue satisfactoria, puesto que el sistema cumplió con los requerimientos que el CSO tenía y está alineado a la forma en que manejaban manualmente los procesos de la gestión de Administración Académica y Evaluación Curricular, agregándole más agilidad y eficiencia, y permitiendo al personal docente y administrativo del Colegio realizar sus tareas de una manera simple y rápida.

La interfaz gráfica diseñada para el sistema el usuario la catalogó como moderna, sencilla y práctica, motivo por el cual fue aceptada inmediatamente y la adaptación a dicha interfaz ha sido inmediata y no requirió ningún tipo de cambio. Todas las funcionalidades que han sido implementadas en el sistema de gestión académica y evaluación curricular vía web, según el usuario del mismo cumplen a cabalidad los procesos académicos que ellos realizan cada año lectivo, razón por la cual, el usuario se ha podido acomodar rápidamente al uso de la aplicación y no ha generado “*oposición al cambio*” al optar por un medio automático de realizar sus tareas diarias, por dicha razón las funcionalidades no requirieron ningún cambio.

4.8.1.3 Retroalimentación

El personal académico del Colegio Sultana del Oriente, una vez realizadas las respectivas pruebas e implementación del sistema de gestión académica y evaluación curricular vía web en su ambiente, manifestaron que el sistema se adapta totalmente a los procesos que manejan en el área académica y que se podría hacer factible iniciar una segunda fase, o versión del sistema, que integre otros procesos realizados manualmente por la institución en la actualidad.

4.9 PRUEBAS DEL SISTEMA

Para que el sistema cumpla con todos los requerimientos establecidos en las historias de usuario, la metodología de Extreme Programming recomienda la utilización de pruebas unitarias con ayuda de herramientas especializadas que las automaticen. Con esto se habilita la propiedad colectiva del código al refactorizar continuamente la solución en desarrollo.

Estas pruebas fueron realizadas en un entorno Web con las siguientes características:

Hardware

Tipo	CLIENTE 1	CLIENTE 2
		PC
Procesador	Intel core 2 duo	Pentium 4
Memoria	2 GB	1 GB
Disco	320 GB	180 GB

Tabla 27 Especificación de hardware usado en Pruebas Unitarias

Red

Conexión	Banda Ancha
Ancho de Banda	512 Kbps
Protocolo	TCP/IPv4

Tabla 28 Especificación de conexión de red utilizada en pruebas

Sistema Operativo / Software

	SERVIDOR	CLIENTE
Sistema Operativo	Linux	Windows XP
Herramientas	APPServer (Apache, MySQL, PHP)	Google Chrome Developer Tools

Navegador Web	Mozilla Firefox 10.0	Mozilla Firefox 10.0
----------------------	----------------------	----------------------

Tabla 29 Especificación del Sistema Operativo y Software utilizado en las Pruebas

Uso de Marcos o Frames

No se utilizan marcos o frames dentro de la estructura de diseño de las páginas que corresponden al sistema. Se propone una hoja de estilos en cascada (CSS versión 3) robusta que se adapta a resoluciones comunes de pantalla evitando el uso de dichos elementos.

Uso de Imágenes en Background

El sistema utiliza imágenes de fondo para el menú y para su fondo general de dimensiones reducidas así como un peso reducido (aprox. 2 KB) con repetición.

4.9.1 ELEMENTOS GRÁFICOS

Peso de las imágenes

Con el propósito de comprobar el peso de las imágenes que constan dentro de la diagramación de la página del sistema, se verificó estas imágenes una por una comprobando que al no tener un tamaño excesivo o un formato no adaptable a web, no genera carga excesiva al momento de ser llamada desde el hosting del CMP.

Cabe mencionar que la mayoría de las imágenes están estandarizadas para las diferentes páginas, utilizando el formato *PNG* con un peso no mayor a 50 KB.

Formato

Los formatos utilizados dentro del sistema son los siguientes:

Función	Formato
Fondos	PNG
Botones	PNG

Iconos	PNG
Menús	PNG

Tabla 30 Formatos de imágenes utilizados

Ubicación

Todas las imágenes que se requiere en el sistema están ubicadas dentro de la carpeta [*dominio*]/*images* como estándar para la ubicación de íconos o botones que se necesite para el sistema.

4.9.1.1 Estándares Esenciales de Navegabilidad

Compatibilidad con Navegadores Web

Los navegadores Web considerados como los más utilizados¹⁴ y a los cuales se recurrieron para esta prueba son:

- Microsoft Internet Explorer (versión 9)
- Mozilla Firefox (versión 10)
- Google Chrome (versión 5.0.375)
- Opera (versión 10.61)
- Apple Safari (versión 5.0.1)

¹⁴ http://www.w3schools.com/browsers/browsers_stats.asp

Microsoft Internet Explorer



Ilustración 17 Prueba de compatibilidad con Microsoft Internet Explorer

La interfaz de usuario sufre pequeñas modificaciones de estilos, debido a que Internet Explorer, en la versión actual, no soporta hojas de estilo escritas en CSS versión 3.

Mozilla Firefox



Ilustración 18 Prueba de compatibilidad con Mozilla Firefox



Ilustración 19 Prueba de compatibilidad con Google Chrome

Opera



Ilustración 20 Prueba de compatibilidad con Opera

La interfaz de usuario sufre modificaciones de estilos, debido a que Opera, en la versión actual, no soporta ciertas características específicas dentro de las hojas de estilo escritas en CSS versión 3.

Resultados Obtenidos

Las interfaces web asignadas al proyecto no sufren ninguna distorsión de diseño por motivos del fabricante del navegador Web, sino únicamente se perciben cambios leves en detalles específicos, a consecuencia de no adoptar el estándar CSS versión 3.

Facilidad de Lectura

Durante el desarrollo se revisó exhaustivamente la correcta lectura de los contenidos de cada una de las páginas que contiene el sistema, eliminando errores tipográficos y posibles incompatibilidades de codificación o caracteres ilegibles.

Se comprobó también muy cuidadosamente que el lenguaje predeterminado a mostrar sea español latinoamericano en cada una de las páginas, principalmente corrigiendo errores en caracteres especiales tal como tildes y de la letra ñ en el despliegue.

Funcionalidad

La funcionalidad fue probada en cada iteración con la secretaria actual del Colegio Sultana del Oriente, y el rector de la institución.

Como usuarios principales del sistema, estas personas se encargaron de realizar simulaciones de los procesos que se manejan a diario en el CSO, con el propósito de verificar que cada uno de estos procesos sea automatizado de manera similar a las prácticas manuales que realizaban.

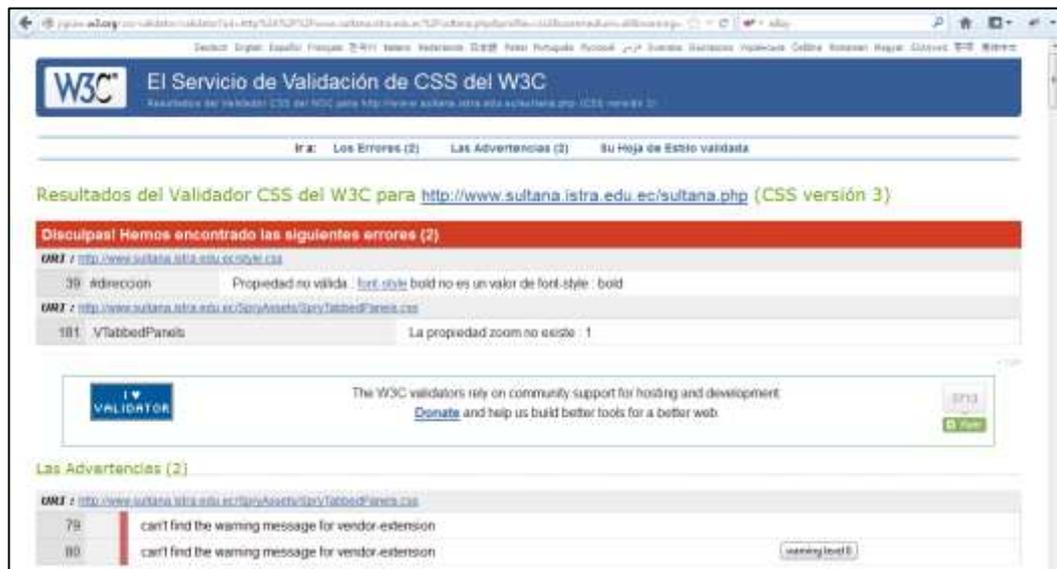
En cada una de las pruebas se obtuvieron refinaciones a realizar en cada una de las interfaces y reportes que se requirieron.

Validación

Para validar código HTML y CSS se disponen herramientas en línea proporcionadas por el consorcio W3C (World Wide Web Consortium)¹⁵, tal como se describió en el Capítulo III.

¹⁵ <http://jigsaw.w3.org/css-validator/>

Debido a que estos validadores se encargan de probar exclusivamente el buen uso de los lenguajes HTML y CSS, únicamente se pudo comprobar la validez de la hoja de estilo mediante este método, obteniendo lo siguiente:



CAPITULO V

METODOLOGIA

5.1 TIPO DE ESTUDIO

La presente investigación de acuerdo al propósito o las finalidades perseguidas se caracteriza de la siguiente manera:

Exploratorio.-Cuerpo de la investigación con el fenómeno para conocerlo o detallarlo tipo pilotaje

Descriptivo.- caracterizar cada variable, manifestaciones o comportamiento de las variables

Asociación de variables.- Medir el impacto de la variable independiente sobre la variable dependiente.

5.2 POBLACIÓN Y MUESTRA

Población

Los individuos que serán objeto de estudio en el presente trabajo de investigación son 120 estudiantes del sexto curso del colegio Particular a Distancia Sultana del Oriente especialidad Informática durante el año lectivo 2011-2012, los 6 docentes y 2 autoridades.

Muestra

En vista de que se trata de una población pequeña no se calcula una muestra por lo tanto se trabajará con el universo en su totalidad.

5.3 OPERACIONALIZACIÓN DE VARIABLES

Variable	Tipo	Definición Conceptual	Dimensiones	Indicadores
Sistema de Evaluación Curricular	Independiente	Software que cumpla con todas las condiciones para el cumplimiento de la evaluación curricular	Software	Funcionalidad. Fiabilidad Usabilidad Eficiencia Mantenibilidad Portabilidad

Mejor control del seguimiento en el cumplimiento del plan anual de contenido.	Dependiente	Cumplimiento del plan curricular anual en el tercer año de bachillerato Informático, según los objetivos académicos	Nivel del logro Objetivos Académicos	Efectividad Tiempo Usabilidad Disponibilidad Producción
---	-------------	---	---	---

Tabla 31 Operacionalización de variables

5.4 PROCEDIMIENTOS

5.4.1 TÉCNICA DE INVESTIGACIÓN

Las técnicas e instrumentos que se utilizarán para la recolección de la información y datos; fundamentalmente son las siguientes:

- Entrevistas
- Encuestas
- Observación

Entrevistas: Dirigidas a los docentes, empleados y alumnado del colegio.

Encuestas: Para conocer las consideraciones de los docentes y estudiantes sobre la organización del sistema de evaluación curricular mediante la WEB.

Observación: Para verificar la realización de la actividad conjunta referida a la transmisión de datos vía WEB.

5.4.2 INSTRUMENTOS DE RECOLECCIÓN DE DATOS

Los instrumentos que se emplearán para la recolección de datos en esta investigación serán:

- Cuestionario de Aceptabilidad del Sistema.
- Guía de Entrevista de recolección de requerimientos de Software
- Ficha de eficiencia de procesos por parte de los usuarios del sistema.

Esto será dirigido y aplicado a los administrativos y docentes del plantel para de esta manera recolectar datos fundamentales para el desarrollo del sistema, y además con estos datos poder hacer un análisis de la hipótesis planteada.

5.5 PROCESAMIENTO Y ANÁLISIS

Para la presente tesis de acuerdo a los objetivos planteados se utilizarán los siguientes modelados y análisis orientados al desarrollo e Ingeniería de Software.

- Revisión crítica de la información recogida; es decir limpieza de información defectuosa: contradictoria, incompleta, no pertinente.
- Tabulación o cuadros según variables de cada hipótesis: cuadros de una sola variable, cuadro con cruce de variables.
- Manejo de información (reajuste de cuadros con casillas vacías o con datos tan reducidos cuantitativamente, que no influyen significativamente en los análisis).
- Estudio estadístico de datos para presentación de resultados en base a la estadística descriptiva por medio de gráficos para su análisis e interpretación respectiva.

5.6 HIPÓTESIS

La implantación de un Sistema de Evaluación Curricular para el Bachillerato Informático del colegio Sultana del Oriente de la ciudad de Macas en entorno Web, aplicando la norma ISO/IEC 9216 incidirá en el cumplimiento del plan anual curricular.

5.7 RESULTADOS

5.7.1 VARIABLE INDEPENDIENTE

DETERMINACIÓN DE PARÁMETROS DE COMPARACIÓN

Para realizar el estudio comparativo de las técnicas de Cumplimiento de la Planificación Anual se tomó en consideración los parámetros necesarios, que permitirán evaluar las cualidades o falencias de cada una de las técnicas escogidas para medir el cumplimiento de los objetivos y según estos compararlos al final y tomar decisiones.

Los parámetros han sido basados en la nueva del de Educación, de sugerencias de docentes en el área y técnicas para elaboración y mejoramiento del software, estos son:

- Funcionalidad
- Fiabilidad
- Usabilidad
- Eficiencia
- Mantenibilidad
- Portabilidad.

1. **Funcionalidad** : Es la capacidad de un producto software de satisfacer los requisitos funcionales prescritos y las necesidades implícitas de los usuarios.
2. **Fiabilidad** : Es la capacidad de un producto software de mantener su nivel de desempeño, bajo condiciones establecidas, por un periodo de tiempo.
3. **Usabilidad**: Es la capacidad de un producto software de ser comprendido, aprendido, usado, atractivo y conforme con las reglamentaciones y guías de usabilidad.
4. **Eficiencia**: Es la capacidad de un producto software de proporcionar un rendimiento apropiado, de acuerdo a la cantidad de recursos usados bajo condiciones establecidas.
5. **Mantenibilidad**: Es la capacidad de un producto software para ser modificado. Las modificaciones pueden incluir correcciones, mejoras o adaptación del software a cambios en el entorno, en los requisitos o en las especificaciones funcionales.
6. **Portabilidad**: Es la capacidad de un producto software de ser transferido

De un ambiente a otro. Nota: El ambiente puede ser organizacional, de software o de hardware.

PROCESAMIENTO DE LA INFORMACION

Para el análisis de la variable independiente, se realizó mediante pruebas con todos los indicadores propuestos, los cuales son:

- Manualmente
- Mediante Sistema
- Mediante Sistema con la Norma ISO 9216

La obtención de un software con calidad implica la utilización de metodologías o procedimientos estándares para el análisis, diseño, programación y prueba del software que permitan uniformar la filosofía de trabajo, en aras de lograr una mayor confiabilidad, mantenibilidad y facilidad de prueba, a la vez que eleven la productividad, tanto para la labor de desarrollo como para el control de la calidad del software.

Los requisitos del software son la base de las medidas de calidad. La falta de concordancia con los requisitos es una falta de calidad.

5.8 ANÁLISIS COMPARATIVO DE LAS TÉCNICAS PARA EL CUMPLIMIENTO DE LA PANIFICACION ANUAL CURRICULAR

VARIABLE INDEPENDIENTE

Se dará a la escala cualitativa los siguientes valores cuantitativos:

Tablas con valores de Si y No:

- Si, tendrá el valor de 1
- No, tendrá el valor de 0

INDICADOR 1: Funcionalidad

Es la capacidad de un producto software de satisfacer los requisitos funcionales prescriptos y las necesidades implícitas de los usuarios.

PARAMETROS		TECNICAS		
		MANUAL	SISTEMA	NORMA ISO
Funcionalidad	Adecuación	SI	SI	SI
	Exactitud	NO	NO	SI
	Seguridad de Acceso	NO	SI	SI
	Cumplimiento funcional	SI	SI	SI

PARAMETROS		TECNICAS		
		MANUAL	SISTEMA	SISTEMA ISO
Funcionalidad	Adecuación	1	1	1
	Exactitud	0	0	1
	Seguridad de Acceso	0	1	1
	Cumplimiento funcional	1	1	1
TOTAL		2	3	4

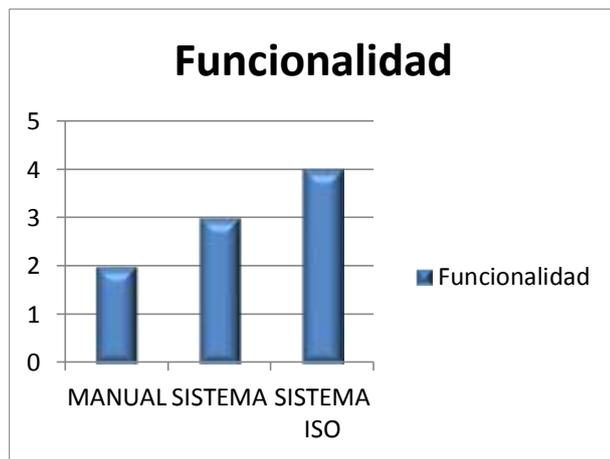


Gráfico 1 Resultados-funcionalidad

Interpretación

Adecuación.- Estamos seguros de que nuestro Software desempeñará perfectamente los requerimientos y necesidades de gestión del colegio a Distancia Sultana del Oriente. Aun así de ser necesario se pueden realizar adecuaciones según los requerimientos bajo una máxima: **CUMPLIMIENTO DE NORMA**

Este trabajo presenta un enfoque práctico para la elección y adecuación de software Open Source de aplicación a contextos específicos. La propuesta está inspirada en metodologías ágiles de desarrollo de software y toma de éstas un conjunto de características que lo hacen práctico, fácil y a la vez desafiante. Este método ha sido aplicado a un proyecto real, cuyo software está finalizado y en etapa de producción, antes se llevaba este trabajo de forma Manual simplemente la diferencia radicaba en el tiempo.

Exactitud.- Al momento de realizar juntas directivas para la toma de decisiones no se podía contar con datos exactos ni reportes de lo que pasaba en realidad con la planificación anual, pero ahora con este Sistema se puede tener reportes de todo lo que paso con los temas de estudio y saber con exactitud las razones por las cuales se desarrollaron o no conjuntamente entre profesor y alumno.

Seguridad de Acceso.-Para este propósito siempre habido problemas de la manera manual, ya que no hay seguridad para acceso de información y muchas de las veces se ha perdido o confundido puesto que a estas encuestas llevadas de manera manual tienen acceso muchas personas de la parte directiva incluyendo los docente, que nada tienen que ver al momento de la toma de decisiones, con el sistema se ha podido controlar el acceso a la información y dar prioridades a los usuarios del mismo.

Cumplimiento funcional.- Esta tarea se la ha llevado muy bien tanto manualmente como con el Sistema.

INDICADOR 2: Fiabilidad

Es la capacidad de un producto software de mantener su nivel de desempeño, bajo condiciones establecidas, por un periodo de tiempo.

PARAMETROS		TECNICAS		
		MANUAL	SISTEMA	NORMA ISO
Fiabilidad	Madurez	NO	SI	SI
	Tolerancia a fallos	NO	NO	SI
	Capacidad de recuperación	NO	SI	SI
	Cumplimiento de la fiabilidad	NO	SI	SI

PARAMETROS		TECNICAS		
		MANUAL	SISTEMA	NORMA ISO
Fiabilidad	Madurez	0	1	1
	Tolerancia a fallos	0	0	1
	Capacidad de recuperación	0	1	1
	Cumplimiento de la fiabilidad	0	1	1
TOTAL		0	3	4

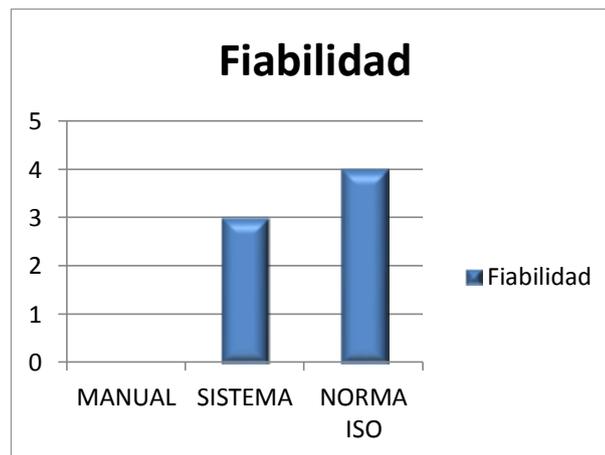


Gráfico 2 Resultados-Fiabilidad

Interpretación

Madurez.- Subcaracterística de fiabilidad, que indica la frecuencia con que ocurren los fallos los cuales se pueden detectar tanto de manera manual como con el sistema.

Tolerancia a fallos.- Subcaracterística de fiabilidad, que indica el grado en que el sistema de evaluación mantiene un nivel de respuesta ante fallos del sistema o interfaces.

Capacidad de recuperación.- De forma manual cuando se perdía algún documento se hacía muy difícil recuperarlo por no decirlo imposible y no había quien se haga responsable del mismo, ahora con el sistema que gracias a una base de datos y respaldos se puede recuperar la información de manera eficiente.

Cumplimiento de la fiabilidad.- Con el Sistema de Evaluación Curricular se puede tener fe de que son datos reales y confiables.

INDICADOR 3: Usabilidad

Es la capacidad de un producto software de ser comprendido, aprendido, usado, atractivo y conforme con las reglamentaciones y guías de usabilidad.

PARAMETROS		TECNICAS		
		MANUAL	SISTEMA	NORMA ISO
Usabilidad	Capacidad de ser entendido	SI	SI	SI
	Capacidad de ser aprendido	SI	SI	SI
	Capacidad para ser operado	SI	SI	SI
	Capacidad de atracción	NO	SI	SI
	Cumplimiento de la usabilidad	SI	SI	SI

PARAMETROS		TECNICAS		
		MANUAL	SISTEMA	NORMA ISO
Usabilidad	Capacidad de ser entendido	1	1	1
	Capacidad de ser aprendido	1	1	1
	Capacidad para ser operado	1	1	1
	Capacidad de atracción	0	1	1
	Cumplimiento de la usabilidad	1	1	1
TOTAL		4	5	5

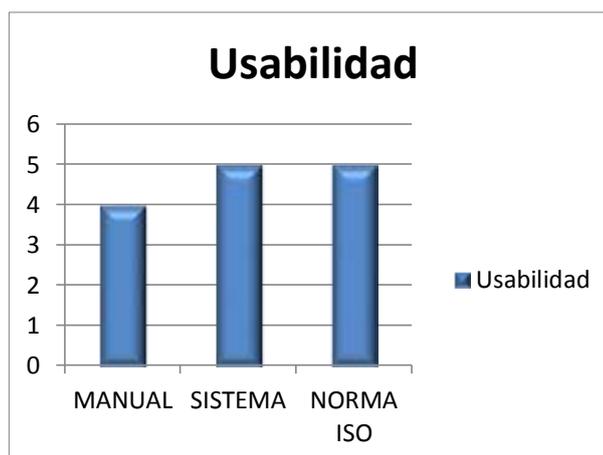


Gráfico 3 Resultados-Usabilidad

Interpretación

Capacidad de ser entendido.- Subcaracterística de facilidad de uso, que indica las características del software que influyen en el esfuerzo del usuario para reconocer el concepto lógico y su aplicación, en la forma manual no hay mucha diferencia.

Capacidad de ser aprendido.- Al momento de llevar la planificación de manera manual siguen ciertos parámetros, los cuales han sido incluidos en el sistema, es decir hay profesores que han decidido llevar su planificación anual manualmente

y al mismo tiempo con el sistema para ir comparando y darse cuenta cual es la mejor alternativa.

Capacidad para ser operado.- Subcaracterística de facilidad de uso, que indica las características del software que influyen en el esfuerzo del usuario para operar y control operacional.

Capacidad de atracción.-Subcaracterística de facilidad de uso, que indica las características del sistema de evaluación que influyen en la satisfacción de los deseos del usuario y las preferencias a través de servicios, comportamiento y presentación más allá de la demanda actual. En cambio de la forma manual no se puede satisfacer esta característica.

Cumplimiento de la usabilidad.-El sistema está de acuerdo con los requerimientos de los usuarios, y estos tienen su origen en el instante en que se llevaba de manera manual las planificaciones.

INDICADOR 4: Eficiencia

Es la capacidad de un producto software de proporcionar un rendimiento apropiado, de acuerdo a la cantidad de recursos usados bajo condiciones establecidas.

PARAMETROS		TECNICAS	
		MANUAL	SISTEMA
Eficiencia	Comportamiento temporal(tiempo de respuesta)	NO	SI
	Utilización de recursos	SI	SI
	Cumplimiento de la eficiencia	NO	SI
			SI
			SI
PARAMETROS		TECNICAS	

		MANUAL	SISTEMA
Eficiencia	Comportamiento temporal(tiempo de respuesta)	0	1
	Utilización de recursos	1	1
	Cumplimiento de la eficiencia	0	1
TOTAL		1	3

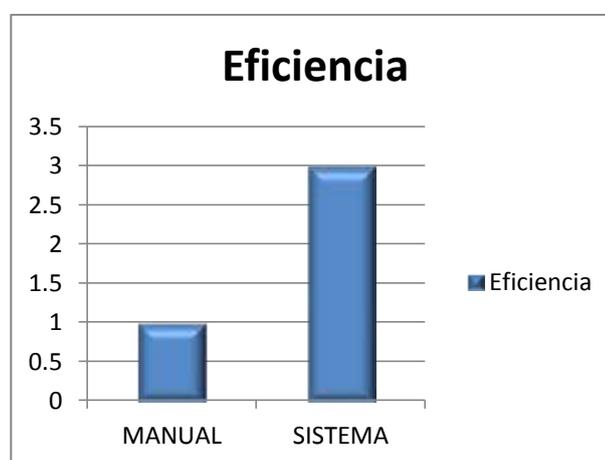


Gráfico 4 Resultados-Eficiencia

Interpretación

Comportamiento temporal.- El sistema de evaluación curricular que posee actualmente el Colegio, tiene tiempos de respuesta muy avanzada y además se podrá obtener reportes al instante lo que jamás pasaba de esa manera en la forma manual por lo que se ha podido optimizar recursos.

Utilización de recursos.- Subcaracterística de eficiencia, que indica las características del sistema de evaluación curricular que influyen en el número de recursos usados, y la duración de su uso, cuando se lleva a cabo.

Cumplimiento de la eficiencia.- Conjunto de características que determinan la relación entre el nivel de rendimiento del software y el número de recursos usados, bajo ciertas condiciones dadas.

INDICADOR 5: Mantenibilidad

Es la capacidad de un producto software para ser modificado. Las modificaciones pueden incluir correcciones, mejoras o adaptación del software a cambios en el entorno, en los requisitos o en las especificaciones funcionales.

PARAMETROS		TECNICAS		
		MANUAL	SISTEMA	NORMA ISO
Mantenibilidad	Capacidad de ser analizado	SI	SI	SI
	Capacidad de ser cambiado	SI	NO	SI
	Estabilidad	NO	SI	SI
	Capacidad de ser probado	NO	SI	SI
	Cumplimiento de la mantenibilidad	NO	SI	SI

PARAMETROS		TECNICAS		
		MANUAL	SISTEMA	NORMA ISO
Mantenibilidad	Capacidad de ser analizado	1	1	1
	Capacidad de ser cambiado	1	0	1
	Estabilidad	0	1	1
	Capacidad de ser probado	0	1	1
	Cumplimiento de la mantenibilidad	0	1	1
TOTAL		2	4	5



Gráfico 5 Resultados-Mantenibilidad

Interpretación

Capacidad de ser analizado.- Subcaracterística de mantenimiento, que indica la cantidad de esfuerzo por el sistema requerido para diagnosticar la causa de un fallo. En la parte manual también se puede diagnosticar un error y corregirlos en el transcurso del año lectivo.

Capacidad de ser cambiado.- Subcaracterística de mantenimiento, que indica la cantidad de esfuerzo requerido para una modificación o borrado de un defecto. Al hacerlos de manera manual se lo ha hecho mediante formularios que paulatinamente que ha pasado el tiempo se han ido también cambiando los formatos.

Estabilidad.- Indica el volumen de riesgos de efectos inesperados tras una modificación. En la parte manual no se puede medir esto ya que es muy difícil detectar los errores y mucho menos corregirlos a tiempo.

Capacidad de ser probado.- El sistema se ha ido probando con todos los docentes y estudiantes para poder llegar a tener un programa de calidad que satisfaga todas las exigencias de los usuarios, pero cuando se lo ha realizado de forma manual siempre nos hemos encontrado dificultades ya que se ha intentado llenar las encuestas mediante papel, y no ha sido satisfactorio por lo que no hemos tenido resultados buenos.

Cumplimiento de la mantenibilidad.- con el presente sistema se puede cumplir con la mantenibilidad para detectar los daños a tiempo para poder decidir. En cambio en la parte manual se podía detectar las fallas de la evaluación manual pero era muy tarde cuando ya se detectaba que había una falencia por lo que había que esperar al año siguiente para adaptar las nuevas recomendaciones y modificaciones.

INDICADOR 6: Portabilidad

Es la capacidad de un producto software de ser transferido de un ambiente a otro. El ambiente puede ser organizacional, de software o de hardware.

Es la capacidad que posee un sistema de información que le permite funcionar en diferentes plataformas ya sean hardware o de software.

PARAMETROS		TECNICAS		
		MANUAL	SISTEMA	NORMA ISO
Portabilidad	Adaptabilidad	SI	NO	SI
	Instalabilidad	NO	SI	SI
	Coexistencia	NO	SI	SI
	Capacidad de reemplazar	NO	SI	SI
	Cumplimiento de la portabilidad	NO	SI	SI

PARAMETROS		TECNICAS		
		MANUAL	SISTEMA	NORMA ISO
Portabilidad	Adaptabilidad	1	0	1
	Instalabilidad	0	1	1
	Coexistencia	0	1	1
	Capacidad de reemplazar	0	1	1
	Cumplimiento de la portabilidad	0	1	1
TOTAL		1	4	5

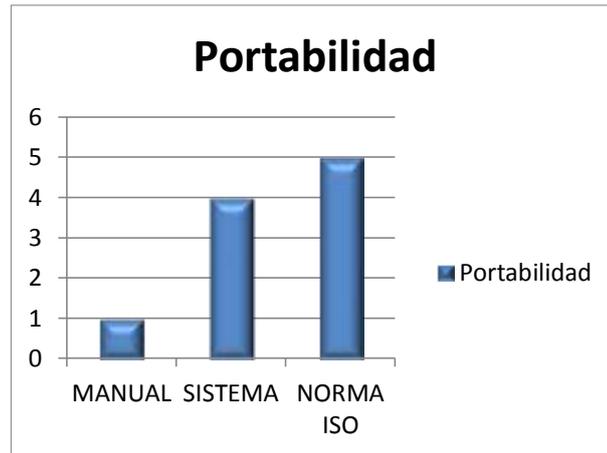


Gráfico 6 Resultados-Portabilidad

Interpretación

Adaptabilidad.- Indica las características de este sistema que influyen en las posibilidades de adaptación a diferentes entornos especificados, sin realizar otras acciones que las indicadas para este propósito. Ya que este programa está diseñado con la norma 9216 no tendremos ningún problema al momento de adaptarlo en otros ambientes.

Instalabilidad.- Se refiere a la capacidad del software de defenderse de las acciones anormales que llevan al sistema a un estado no deseado o por lo menos no previsto, causando un comportamiento inesperado, indeseado y posiblemente erróneo. El software de hoy, está en capacidad de analizar los datos que recibe para hacer cumplir requerimientos o condiciones del software y enfrentar de la mejor manera los errores cometidos por un usuario al utilizar la aplicación. Es importante resaltar, que la solidez no siempre es generada por la digitación inapropiada del usuario, sino también por un mal procesamiento o un mal encadenamiento de procesos. El resultado de un proceso, aunque sea correcto, puede estar fuera de los límites permitidos en los parámetros del módulo que lo recibe y si este módulo no controla los parámetros que le entran caerá en un estado inesperado.

Coexistencia.- Nos Indica la capacidad del software de coexistir con otro software independiente en un entorno común compartiendo recursos. Esto solo se

cumple en este sistema ya que de manera manual no se pueden llevar dos metodologías simultáneamente.

Capacidad de reemplazar.- Subcaracterística de portabilidad, que indica las características del software que influyen en la posibilidad y esfuerzo requerido para usarlo en lugar de otro software en el mismo entorno.

Cumplimiento de la portabilidad.- Se refiere a que los resultados de operaciones sean acordes al comportamiento del mundo real desde todos los estados y condiciones permitidos por la aplicación, es decir, el programa debe reflejar la realidad. Un programa es inconsistente si presenta respuestas erróneas en algunos casos. Una mala especificación de rangos en un dominio sobre los cuales realizan diferentes operaciones matemáticas puede llevar a que algunos cálculos se realicen dentro de límites inapropiados, obteniéndose resultados erróneos. Otro caso de inconsistencia se presenta cuando ocurren eventos que paran abruptamente la ejecución del programa, sólo un sistema de calidad podrá conservar datos consistentes después de una falla.

PARAMETROS		TECNICAS		
		MANUAL	SISTEMAS	SISTEMA ISO
Funcionalidad	Adecuación	X	X	X
	Exactitud			X
	Seguridad de Acceso		X	X
Fiabilidad	Madurez	X	X	X
	Tolerancia a fallos			X
	Capacidad de recuperación		X	X
	Cumplimiento de la fiabilidad		X	X
Usabilidad	Capacidad de ser entendido		X	X
	Capacidad de ser aprendido	X	X	X
	Capacidad de atracción	X	X	X
	Cumplimiento de la usabilidad		X	X
Eficiencia	Comportamiento temporal		X	X
	Utilización de recursos	X	X	X
	Cumplimiento de la eficiencia			X
Mantenibilidad	Capacidad de ser analizado	X	X	X

ad	Capacidad de ser cambiado	X		X
	Estabilidad		X	X
	Capacidad de ser probado		X	X
	Cumplimiento de la manten		X	X
Portabilidad	Adaptabilidad	X		X
	Instalabilidad		X	X
	Coexistencia		X	X
	Capacidad de reemplazar		X	X
	Cumplimiento de la portabilidad		X	X
TOTAL		11	19	26

5.9 ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

En este trabajo de investigación se ha tratado de presentar de manera concisa una comparación de 3 técnicas: Manual, Mediante Sistema Anterior y Sistema Actual mediante la Norma ISO 9216 para el mejoramiento en el control del cumplimiento de la Evaluación Curricular y así poder verificar logros y metas propuestos al inicio del año.

Con la técnica manual se tuvo éxito durante mucho tiempo ya que no existía la computadora o no se tenía el conocimiento necesario para su manejo. Se llevaba las planificaciones en un libro y en cada clase realizada se anotaba en el registro cual fue la clase dada, y en observaciones se ponía hasta donde avanzaba. Estos avances eran revisados minuciosamente por una junta de área para constatar si se está cumpliendo con las metas planteadas pero no cumplía con el parámetro de Fiabilidad puesto que no se podía hacer un control del cumplimiento curricular de todos los cursos y todos los paralelos, por lo que se optó por hacer un sorteo del 20% de los cursos para ver su cumplimiento. Estos datos tampoco cumplían con el indicador de Eficiencia puesto que los estudiantes no tenían conocimientos de lagunas unidades revisadas dando como resultado un reporte de que no se vieron ciertos temas y no se pudieron tomar ciertas medidas a tiempo. Además se realizaban encuestas trimestrales a los estudiantes para medir el cumplimiento de los temas realizados, sin tener éxito ya que los resultados eran poco reales y no existía colaboración ni de directivos, docentes y estudiantes por lo que resulto una total pérdida de tiempo y recursos.

Posteriormente al implementar un sistema Académico, tenía la opción de realizar encuestas en línea conectándose todos al servidor en el laboratorio de informática lo cual dio resultados excelentes, pero al paso del tiempo se tuvo problemas en la Mantenibilidad ya que era incapaz de ser cambiado y también daba problemas en la Portabilidad ya que se hacía difícil adaptarlo a otros ambientes. Por último el sistema era muy eficiente pero no arrojaba datos exactos puesto que el usuario no tenía una cuenta en este sistema por lo que los estudiantes solamente llenaban las encuestas digitalmente pero sin datos reales. También el sistema no fue desarrollado con software libre.

Por Último se lanzó El sistema Académico en la WEB dando resultados excelentes puesto que tiene mucha adaptabilidad con ambientes web, se puede ir cambiando a medida que pase el tiempo puesto que es software libre. Los datos son exactos y la junta de área puede analizar los resultados y tomar decisiones a medida que avancen los temas de las unidades. Este sistema es muy eficiente y también fiable ya que las encuestas realizadas se actualizan constantemente en la base de datos.

5.9.1 VARIABLE DEPENDIENTE

ANÁLISIS Y PRESENTACIÓN DE RESULTADOS

PROPUESTA METODOLÓGICA DE LA EVALUACION CURICULAR

Para el análisis de la variable dependiente se le dará a la calificación cualitativa un valor como se muestra a continuación:

CALIFICACIÓN	ABREVIATURA	VALOR ASIGNADO
No Existe	NE	0
Malo	M	1
Regular	R	2
Bueno	B	3
Muy Bueno	MB	4
Excelente	E	5

Tabla 32 Correspondencias para Análisis de la Variable Independiente

Escala cuantitativa: del 0 (no existe) al 5 (excelente). Cada número tiene su correspondencia en porcentaje, que es el que se utilizará en el cumplimiento del registro de evaluación del análisis de cada uno de los índices.

VALORACIÓN					
0	1	2	3	4	5
0%	20%	40%	60%	80%	100%

INDICADOR 1: Efectividad

- **Índice:** Eficiencia

Efectividad	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Eficiencia					X							X

Efectividad	MANUAL	SISTEMA
Eficiencia	80%	100%

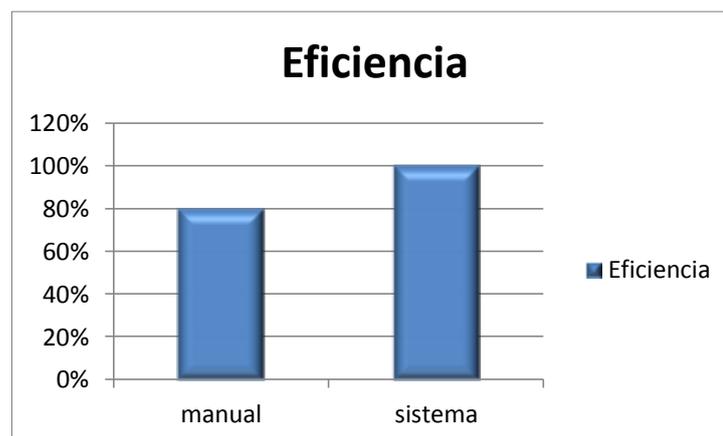


Gráfico 7 Eficiencia

Interpretación

Al comparar la eficiencia del método manual y sistema tenemos como resultado que el método informático tiene mejor funcionamiento y mejores resultados ya que optimiza el rendimiento de los recursos utilizados por un proceso.

- **Índice: Eficacia**

Efectividad	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Eficacia				X								X

Efectividad	MANUAL	SISTEMA
Eficacia	60%	100%

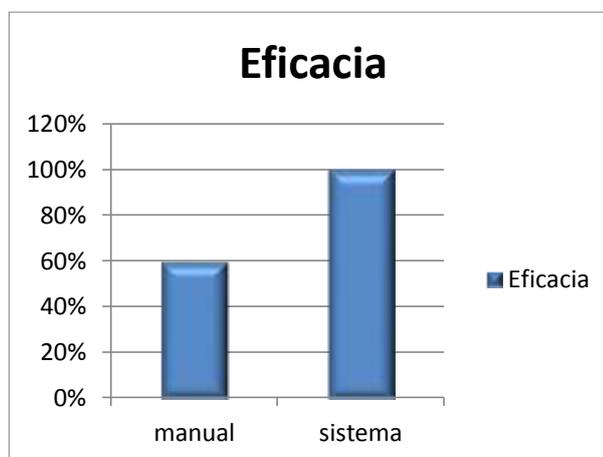


Gráfico 8 Eficacia

Interpretación

Al hacer un análisis sobre cumplimiento de los objetivos propuestos, se enfocan en el qué se debe hacer, para este indicador se deben conocer y definir los requerimientos de los estudiantes del proceso.

- **Índice: Cumplimiento de Efectividad**

Efectividad	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Cumplimiento de Efectividad			X									X

Efectividad	MANUAL	SISTEMA
Cumplimiento de Efectividad	40%	100%

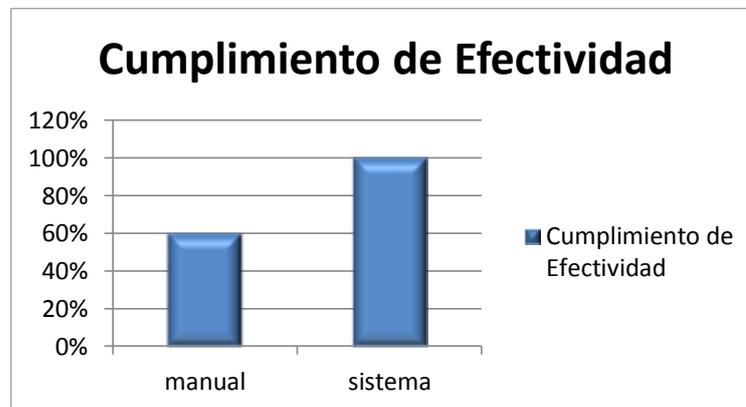


Gráfico 9 Efectividad

Interpretación

Mediante este indicador se puede decir que con el sistema informático hay satisfacción en la comunidad estudiantil así como la parte directiva

INDICADOR 2: Usabilidad

- Índice: Capacidad de ser entendido

Usabilidad	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Capacidad de ser entendido			X									X

Usabilidad	MANUAL	SISTEMA
Capacidad de ser entendido	40%	100%

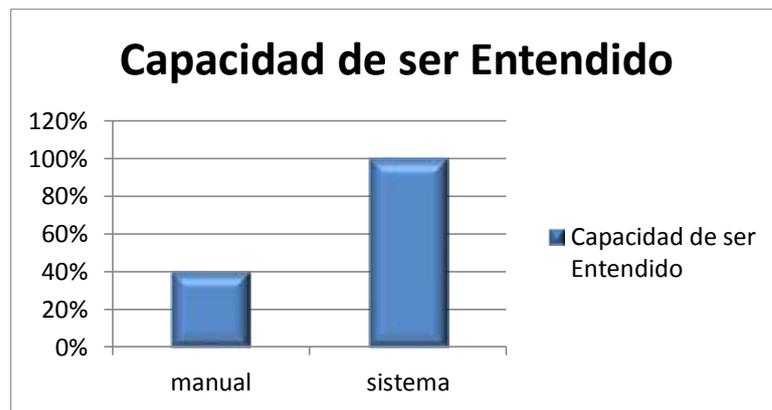


Gráfico 10 Capacidad de ser entendido

Interpretación

Subcaracterística de facilidad de uso, que indica las características del software que influyen en el esfuerzo del usuario para reconocer el concepto lógico y su aplicación, en la forma manual no hay mucha diferencia.

- Índice: Capacidad de ser aprendido

Usabilidad	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Capacidad de ser aprendido				X								X

Usabilidad	MANUAL						SISTEMA					
	60%						100%					

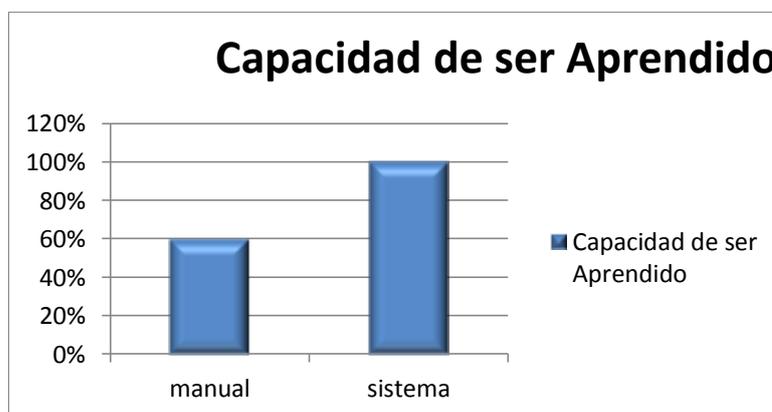


Gráfico 11 Capacidad de ser aprendido

Interpretación

Al momento de llevar la planificación de manera manual siguen ciertos parámetros, los cuales han sido incluidos en el sistema, es decir hay profesores que han decidido llevar su planificación anual manualmente y al mismo tiempo con el sistema para ir comparando y darse cuenta cual es la mejor alternativa.

- Índice: Capacidad de ser operado

Usabilidad	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Capacidad para ser operado				X								X

Usabilidad	MANUAL						SISTEMA					
	Capacidad para ser operado	60%						100%				

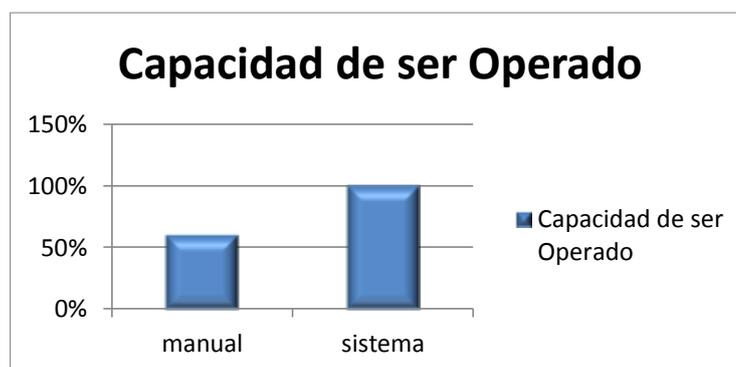


Gráfico 12 Capacidad de ser operado

Interpretación

Subcaracterística de facilidad de uso¹⁶, que indica las características del software que influyen en el esfuerzo del usuario para operar y control operacional.

¹⁶ <http://squac.iti.upv.es/glosario-calidad/task/view/id,31/>

Índice: Capacidad de atracción.

Usabilidad	MANUAL					SISTEMA						
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Capacidad de atracción				X								X

Usabilidad	MANUAL					SISTEMA				
Capacidad de atracción	60%					100%				

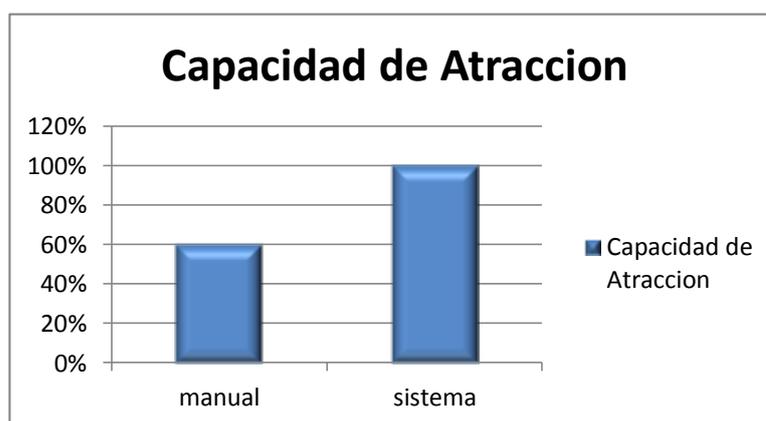


Gráfico 13 Capacidad de abstracción

Interpretación

Subcaracterística de facilidad de uso, que indica las características del sistema de evaluación que influyen en la satisfacción de los deseos del usuario y las preferencias a través de servicios, comportamiento y presentación más allá de la demanda actual. En cambio de la forma manual no se puede satisfacer esta característica.

- Índice: Cumplimiento de Usabilidad

Usabilidad	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Cumplimiento de la usabilidad				X								X

Usabilidad	MANUAL						SISTEMA					
	Cumplimiento de la usabilidad	60%						100%				



Gráfico 14 Cumplimiento de usabilidad

Interpretación

El sistema está de acuerdo con los requerimientos de los usuarios, y estos tienen su origen en el instante en que se llevaba de manera manual las planificaciones.

INDICADOR 3: Disponibilidad

- Índice: Confiabilidad

Disponibilidad	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Confiabilidad				X								X

Disponibilidad	MANUAL						SISTEMA					
	60%						100%					

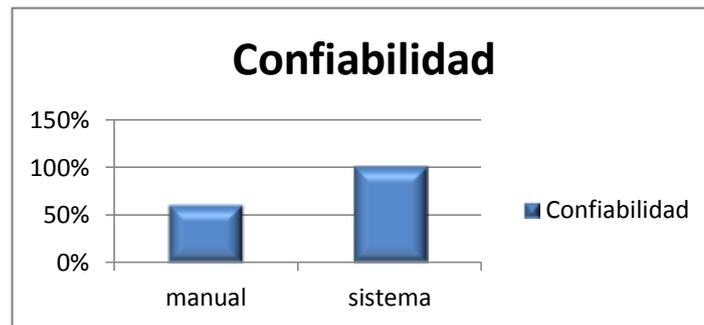


Gráfico 15 Confiabilidad

Interpretación

La confiabilidad tiene muchos significados técnicos diferentes, pero uno de los más amplios es el siguiente: la confiabilidad es la característica de un elemento expresada por la probabilidad de que cumpla sus funciones específicas durante un tiempo determinado, cuando se coloca en las condiciones del medio exterior. La definición también se puede expresar como la probabilidad de que un equipo no falle mientras esté en servicio durante un período dado. Lo que no sucede con la parte manual que solo depende de la persona que esté realizando las encuestas

La confiabilidad como parámetro adaptado al criterio de disponibilidad se define como la probabilidad de que un equipo no falle estando en servicio dentro de un período de tiempo determinado

Índice: Mantenibilidad

Disponibilidad	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Mantenibilidad				X								X

Disponibilidad	MANUAL						SISTEMA					
Mantenibilidad	60%						100%					

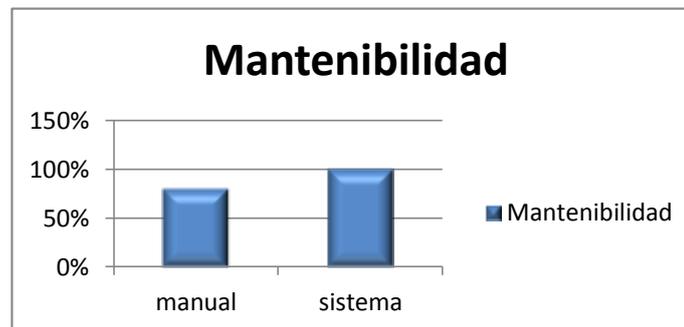


Gráfico 16 Mantenibilidad

Interpretación

El sistema de evaluación curricular en estado de operación debe, en algún momento, dejar de funcionar por un intervalo de tiempo, lo cual implica que ha cambiado a un estado de no-operación. En el caso de que las exigencias de un programa sean de operación continua, toda condición fuera de ésta se considera como una falla. En la práctica, nos interesa que la duración de la falla sea lo más corta posible y, en ese sentido, mientras menos falle un sistema su confiabilidad es mayor.

Las causas de una falla son propias de un sistema y pueden originarse por problemas de diseño, todo esto influye para que el comportamiento de las fallas del sistema de evaluación. Esto no sucede al llevarlo de la manera manual ya que no se depende de que funcione o no el sistema de evaluación.

Una vez detectada la falla lo importante es corregirla y retornar el equipo o sistema al estado de operación normal.

INDICADOR 4: Producción

- Índice: Calidad

Producción	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Calidad				X								X

Producción	MANUAL						SISTEMA					
Calidad	60%						100%					



Gráfico 17 Calidad

Interpretación

La calidad de la evaluación curricular tiene mucho que ver con los resultados obtenidos, y de acuerdo a esto tomar decisiones a tiempo. En lo que se refiere a la evaluación realizada a mano no se obtienen reportes al instante y no se puede confiar de estos resultados.

- **Índice: Costo**

Producción	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Costo				X								X

Producción	MANUAL						SISTEMA					
	Costo						Costo					
	60%						100%					

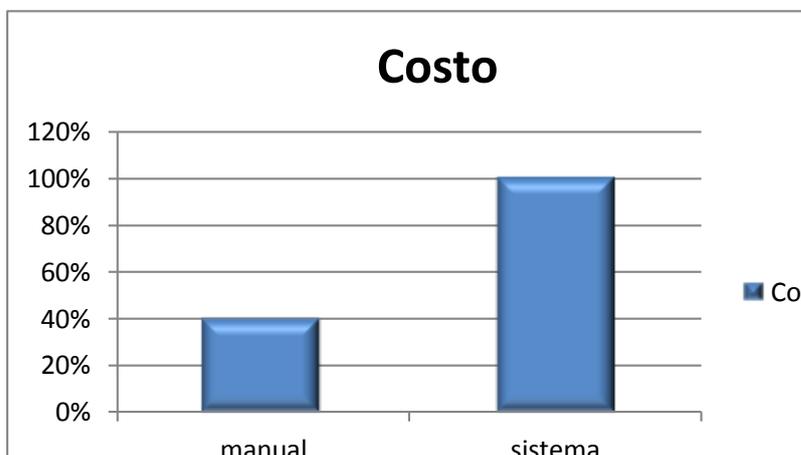


Gráfico 18 Costo

Interpretación

El costo del sistema es elevado, pero mayor era el costo con papel y el tiempo que se demoraba en evaluar y tomar decisiones.

INDICADOR 5: Tiempo

- Índice: Plazo

Tiempo	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Plazo				X								X

Tiempo	MANUAL						SISTEMA					
	60%						100%					
Plazo												

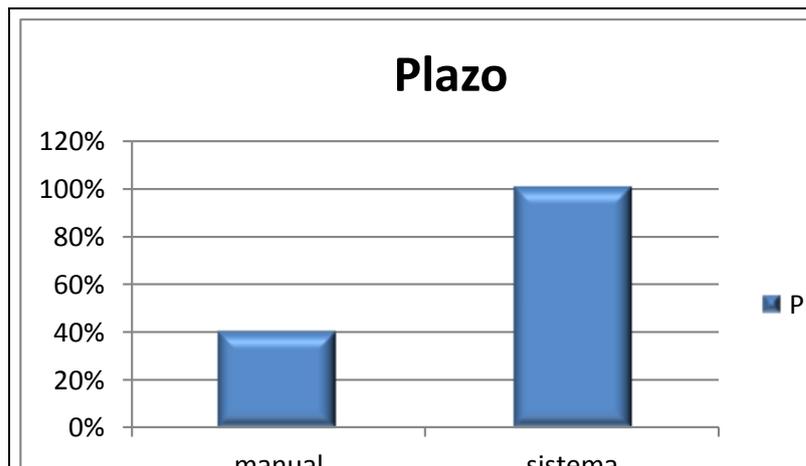


Gráfico 19 Plazo

Interpretación

Cada evaluación tiene un plazo los cuales no se podían visualizar de manera manual ya que se demoraba y pero aun tomar decisiones, tenemos que tiene un cumplimiento del 40% según datos dados por los mismos estudiantes. Con el sistema actual se cumple con reportes en las fechas esperadas.

- **Índice: Factor Tiempo**

Tiempo	MANUAL						SISTEMA					
	NE	M	R	B	MB	E	NE	M	R	B	MB	E
Factor Humano				X								X

Tiempo	MANUAL	SISTEMA
Factor Humano	60%	100%

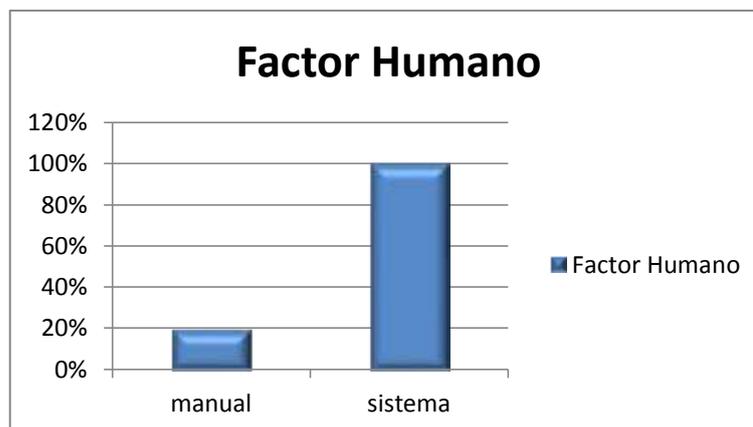


Gráfico 20 Factor Humana

Interpretación

Para hacer la evaluación curricular manualmente se debe sortear determinados cursos para poder alcanzar por lo que no se tiene un reporte claro de lo que está pasando con el cumplimiento de objetivos en las planificaciones, su porcentaje es bajo por esta razón.

INDICADOR	INDICES	TECNICAS	
		MANUAL	SISTEMA
Efectividad	eficiencia	0,80	1,00
	eficacia	0,60	1,00
	Cumplimiento de efectividad	0,40	1,00
Tiempo	Plazo	0,60	1,00
	Factor humano	0,60	1,00
Usabilidad	Capacidad de ser entendido	0,40	1,00
	Capacidad de ser aprendido	0,60	1,00
	Capacidad de atracción	0,60	1,00
	Cumplimiento de la usabilidad	0,60	1,00
Disponibilidad	Confiabilidad	0,60	1,00
	Mantenibilidad	0,60	1,00
Producción	Calidad	0,60	1,00
	Costo	0,60	1,00

5.10 PRUEBA DE LA HIPOTESIS

En investigación científica muchas de las veces es imposible hacer un estudio exhaustivo de los elementos de la población o lo que se conoce como un estudio poblacional o censo, para este caso, se hace necesario valerse de las técnicas de muestreo para tomar sólo una parte la cual debe ser representativa de la población de estudio. En este punto vale aclarar que el objetivo no corresponde a realizar diseños de muestreo para investigación científica sino en conocer la distribución que sigue la media muestral siendo necesario conocer algunos tópicos iniciales de muestreo.

Aparece de manera natural al realizar la prueba t de Student para la determinación de las diferencias entre dos medias muestrales y para la construcción del intervalo de confianza para la diferencia entre las medias de dos poblaciones cuando se desconoce la desviación típica de una población y ésta debe ser estimada a partir de los datos de una muestra.

En general, en los problemas que se presentan habitualmente, existe una población de observaciones cualesquiera, de la cual tomamos una muestra aleatoria, por medio de la cual intentamos conocer todo lo que sea posible acerca de la

población de la cual fue extraída. El promedio de la muestra de n elementos pertenece a la distribución de promedios muestrales de la población original. Es decir, que el promedio de la muestra que obtuvimos es uno de los muchos promedios muestrales que se distribuyen alrededor de m con desviación standard.

Por lo tanto, si la muestra es más grande (n mayor), estaremos en una distribución de promedios con desviación standard más pequeña, por lo cual, el promedio de la muestra estará más cerca del promedio del universo. Es por esto que es razonable pensar que el promedio de la muestra es una estimación del promedio del universo.

En la tabla que aparece a continuación se aprecian una encuesta realizada a 30 estudiantes a los cuales se les formula 15 preguntas en las cuales tiene como fondo analizar cual método es mejor para llevar la evaluación curricular tales como: manualmente o mediante sistema.

ENCUESTA	MANUAL(Y)	SISTEMA(X)	X - Y	(X-Y)^2
eficiencia	10	30	20	400
eficacia	15	25	10	100
Cumplimiento de efectividad	5	35	30	900
Plazo	8	25	17	289
Factor humano	4	36	32	1024
Capacidad de ser entendido	20	20	0	0
Capacidad de ser aprendido	16	24	8	64
Capacidad de atracción	30	10	-20	400
Cumplimiento de usabilidad	30	10	-20	400
Confiabilidad	18	22	4	16
Mantenibilidad	24	26	2	4
Calidad	16	24	8	64
Costo	15	25	10	100
Satisfacción	7	33	26	676
Contenidos	10	30	20	400
	228	375	147	3410
			9,8	

Planteamiento de la hipótesis

$$H_0: \mu d \leq 0$$

$$H_1: \mu d > 0$$

Ecuación 1

Nivel de significación

$$\alpha = 0,05$$

Regiones de rechazo y aceptación

La RR está dada por todos los valores t observados tal que:

$$t > t_{0,95; 14}$$

$$t > 1,7613$$

$$Sd^2 = \Sigma di^2 - n \cdot (\bar{e})^2 / n - 1$$

Ecuación 2

$$Sd^2 = 3410 - 15 \cdot (9,8)^2 / 14$$

$$Sd^2 = 140,11$$

$$Sd = 11,86$$

$$t = (\bar{e} - \mu) / (s / \sqrt{n})$$

$$t = (9,8 - 0) / (11,86 / \sqrt{15})$$

$$t = 3,2$$

Como t es 3,2; es mayor que el crítico 1,76 se rechaza H_0 .

Con un 95% de confianza se puede concluir que con el sistema informático los estudiantes están más satisfechos que con el sistema manual.

5.11 DISCUSIÓN

En este primer planteamiento sobresale que un programa educativo es mucho más que un plan de estudios, ya que significa también analizar las formas de gobierno, los mecanismos para designación de las autoridades académicas, asignaturas, contenidos, metas, tecnologías que se aplican a estas, así como una ponderación sobre la normatividad institucional y sobre el conjunto de reglamentos con los que opera, la calificación del personal académico medida en tiempo de dedicación al programa y en sus planificaciones. Importancia particular tiene el proceso de selección de los temas analizados por los estudiantes a partir de una evaluación de los temas de las unidades, así como el estudio y análisis al

final sobre el cumplimiento de metas. De alguna forma se promueve que un buen programa es aquel donde los nuevos bachilleres se incorporan al mundo universitario o de trabajo después de los seis meses de su incorporación.

El éxito de este sistema de evaluación curricular es poseer una arquitectura orientada a módulos está directamente relacionada con la habilidad que tiene el programador para desarrollar módulos que puedan utilizarse en otros sitios y resolver necesidades específicas de varias empresas u organizaciones.

En el desarrollo de esta página se piensa hacer contacto con sitios Web gubernamentales como ministerios de educación o colegios, que contribuyen con módulos que pueden ser reutilizados por otras entidades o cualquier aplicación, lo que facilita e incrementa el valor de la plataforma para sitios Web o aplicaciones Web del sector de la Educación, los costos de desarrollo e implementación son de bajo costo, se puede acceder al código fuente, a la documentación ya que la tecnología es de software libre.

Al momento de integrar contenidos media como vídeos, audio, imágenes y listas de reproducciones en un sitio Web, se puede realizar utilizando los módulos como SWFTools, Embedded Media Field, mp3Player, etc. Estos módulos facilitan el trabajo en este tipo de sitios Web y permite integrar a la plataforma funcionalidades a los usuarios donde pueden compartir, distribuir, calificar el contenido del sitio web, y ayudar al crecimiento del Sistema Académico.

Este sistema de evaluación curricular ofrece gran flexibilidad, potencia, escalabilidad, soporte, seguridad, es altamente configurable, presenta controles de workflow, y cuenta con excelentes controles de accesos de usuarios.

CAPITULO VI

CONCLUSIONES Y RECOMENDACIONES

6.1 CONCLUSIONES

El estudio y aplicación de la norma ISO/IEC 9216 permite aplicar parámetros de calidad como funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad para desarrollar un sistema que cumpla con las necesidades del usuario ya que la norma se enfoca en varios parámetros orientados a la simplicidad y eficiencia al cumplir una tarea.

El sistema permite integrar en los módulos programados en PHP, ficheros de JavaScripts, CSS, los cuales mejoran la funcionalidad tal como rapidez de carga, facilidad de manejo, eficiencia, fiabilidad, portabilidad y apariencia del módulo.

El nuevo enfoque de la evaluación curricular para el seguimiento del control del cumplimiento curricular se refiere a la evaluación concomitante de procesos y resultados. Dirige su atención al trabajo de profesores, investigadores y estudiantes, así como a las condiciones que facilitan u obstaculizan el desempeño de todos estos actores. La finalidad de la evaluación curricular es ahora proporcionar al evaluador un conocimiento sobre el desempeño de los procesos y sus resultados; unos y otros han de ser objeto de análisis, y no sólo para ser juzgados conforme a parámetros previamente establecidos, sino además para comprender mejor las relaciones existentes entre procesos, entre resultados diversos y entre procesos y resultados.

La metodología XP establece seguir un proceso ordenado para la implementación del sistema "sultana" ya que esta metodología permite una programación ágil y ligada a continuas iteraciones y entregas de prototipos ideal para proyectos de no muy largo tiempo y complejidad moderada. Con la programación de entregas se obtuvieron prototipos de los módulos definidas y aprobados en cada entrega para de esta forma no tener que hacer cambios en las mismas al final del proyecto, dejando este aspecto para la fase de mantenimiento.

La Ingeniería de Software aplicada en el proyecto está basada en la entrega de prototipos iterativos establecidos en el análisis de requisitos obtenidos por medio de Historias de Usuario detalladas por los usuarios del sistema, lo cual se adapta convenientemente con la metodología de desarrollo eXtreme Programming.

El empleo de software libre facilita un sistema conveniente para el dueño del proyecto ya que no tendrá necesidad de hacer inversiones en licencias costosas ya que el sistema está programado básicamente en lenguajes de programación HTML, PHP y servidor de base de datos MySQL. Debido a que Notepad++ es de libre distribución y tiene soporte para lenguajes como PHP, HTML y varios más, resulta muy útil para la programación del sistema web "Sultana".

El proyecto propuesto plantea exclusivamente el control del cumplimiento del plan curricular pero esto no quiere decir que en un futuro no se puedan hacer cambios para integrar módulos que permitan evaluar y controlar otros entes como el diseño curricular, el seguimiento académico ya que el sistema admite la implementación modular, permitiendo integrar futuras modificaciones en las Normativas del Ministerio de Educación del Ecuador.

6.2 RECOMENDACIONES

Para cargar o crear un contenido a una página, tener previamente estructurados los contenidos para ser publicados, es decir se debe definir las secciones, categorías y los contenidos que contengan la página.

Cuando instale el Sistema Académico hay que tomar en cuenta que las carpetas tenga todos los permisos de Lectura, Escritura y Ejecución, caso contrario el Sistema no podrá modificar dichos archivos y no se podrá seguir con la configuración del sistema.

Utilizar plantillas en donde exista un mínimo de contenido flash en su página principal, con el fin de que cargue rápidamente el portal en la pantalla.

Como principal aspecto a tomar en cuenta al momento de realizar un sitio web son las seguridades que tendrá la aplicación puesto que en los últimos años se ha incrementado el ataque a los sitios web por parte de terceros o intrusos lo cual podemos contrarrestar con la utilización de varios módulos como Paranoia, Bad Behavior, Captcha, Persistent login, Taxonomy Access Control, esto no quita que el desarrollador pueda integrar sus propias seguridades con el uso de scripts en PHP, lo cual hará un sitio más seguro y robusto.

Se recomienda continuar con la fase de mantenimiento del sistema ya que será necesario enmendar futuras observaciones por parte de los usuarios del sistema.

GLOSARIO

Análisis (analysis)

Proceso de identificación, modelado y descripción de lo que hace un sistema y de cómo trabaja

Aplicación (application)

Programa autónomo Java tal como cualquier programa escrito utilizando un lenguaje de alto nivel. Las aplicaciones se pueden ejecutar desde cualquier computadora con un intérprete Java. Las aplicaciones no están sometidas a las restricciones impuestas los applets de Java. Una clase aplicación debe contener un método main. Se utiliza como sinónimo de programa.

Asignación (assignment)

Almacenamiento de un valor en una variable. La sentencia de asignación es aquella que implementa la asignación y utiliza un operador de asignación

Contenedor (container)

Clase que implementa una estructura de datos que contiene una colección de objetos. Se utiliza también para representar un componente IGU, Interfaz Gráfica de Usuario (GUI; Graphical User Interface) que contiene una colección de otros componentes IGU

Diseño (design)

Actividad de definir como se debe estructurar e implementar un programa.

HTML (Hypertext Markup Language)

Lenguaje en que se escriben los documentos que se utilizan en Internet.

IDE (integrated development)

Software para ayudar a los programadores a escribir código eficientemente.

IGU, Interfaz Gráfica de Usuario (GUI, GraphicalUser Interface)

Una interfaz es un programa que se implementa utilizando componentes AWT tales como cuadros, botones, etiquetas, campos de texto, etc.

Implementación (implementation)

La actividad de escribir, compilar, probar y depurar el código de un programa.

JDK (Java development kit)

Define el API de Java y contiene un conjunto de utilidades de líneas de órdenes tales como Javac (compilador) y Java(interprete).

JavaScript

Un lenguaje de comandos multiplataforma del WWW desarrollado por Netscape Communications. El código de JavaScript se inserta directamente en una página HTML.

WWW (World Wide Web)

Es uno de los servicios más atractivos de Internet. Esta aplicación, cuyo software más utilizado es Netscape, permite transmitir y visualizar imágenes, audio, gráfica y textos a través de la red.

BIBLIOGRAFIA

Páginas web

ENRIQUE J. ABRIL PÉREZ, Calidad de las Aplicaciones Web
<http://zarza.usal.es/~fgarcia/doctorado/iweb/05-7/Trabajos/CalidadAplicWeb>,
7/7/2012.

CALIDAD DEL SOFTWARE, Calidad del software,
<http://www.slideshare.net/lcahuich/calidad-del-software-presentation>, 2/12/2011.

METODOLOGÍA EXTREAM PROGRAMMING, Metodologías de SW,
<http://oness.sourceforge.net/proyecto/html/ch05s02.html>, 6/9/2012.

MARIFI2, Calidad del Software, Tecnologías de Internet,
<http://marifi2.blogspot.es/>, 6/10/2012.

MINISTERIO DE EDUCACIÓN Y CULTURA, Bachillerato Técnico Polivalente,
<http://www.uasb.edu.ec/reforma/paginas/btp/btp13.htm>, 15/4/2012.

SLIDESHARE, Metodología XP, <http://www.slideshare.net/CrisCobol/metodologia-xp-6697221>, 23/05/2012.

THE PHP GROUP, Php, <http://php.net/manual/es/tutorial.requirements.php>,
24/05/2012.

TUTORIAL DE PHP Y MYSQL. Tutorial de php y mysql
www.antisacsor.com/download.php?id=5, 12/04/2012.

DESARROLLOWEB.COM, Manual De Css, <http://www.desarrolloweb.com/manuales/css3.html>, 06/06/2012.

HTMLCINCO, Manual de html5 en español, <http://www.htmlcinco.com/manual-de-html5-en-espanol/> 3/4/2012.

TALLER WEBMASTER, Instalación de AppServ, <http://www.htmlcinco.com/manual-de-html5-en-espanol/>, 23/04/2012.

CREATIVOSON LINE, Manual de Dreamweaver 5 en español,
<http://www.creativosonline.org/blog/manual-de-dreamweaver-cs5-en-espaol.html>,
24/6/2012.

ANEXO A



Ilustración 21 Gestión de Especialidades



Ilustración 22 Gestión de Cursos



Ilustración 23 Gestión de Paralelos



Ilustración 24 Asignación de Docentes

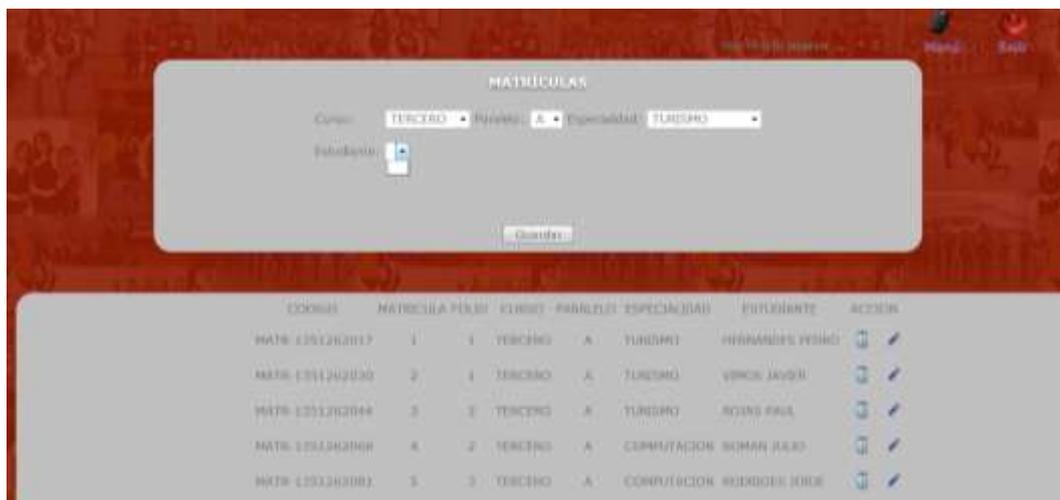


Ilustración 25 Gestión de Matriculas

ANEXO B

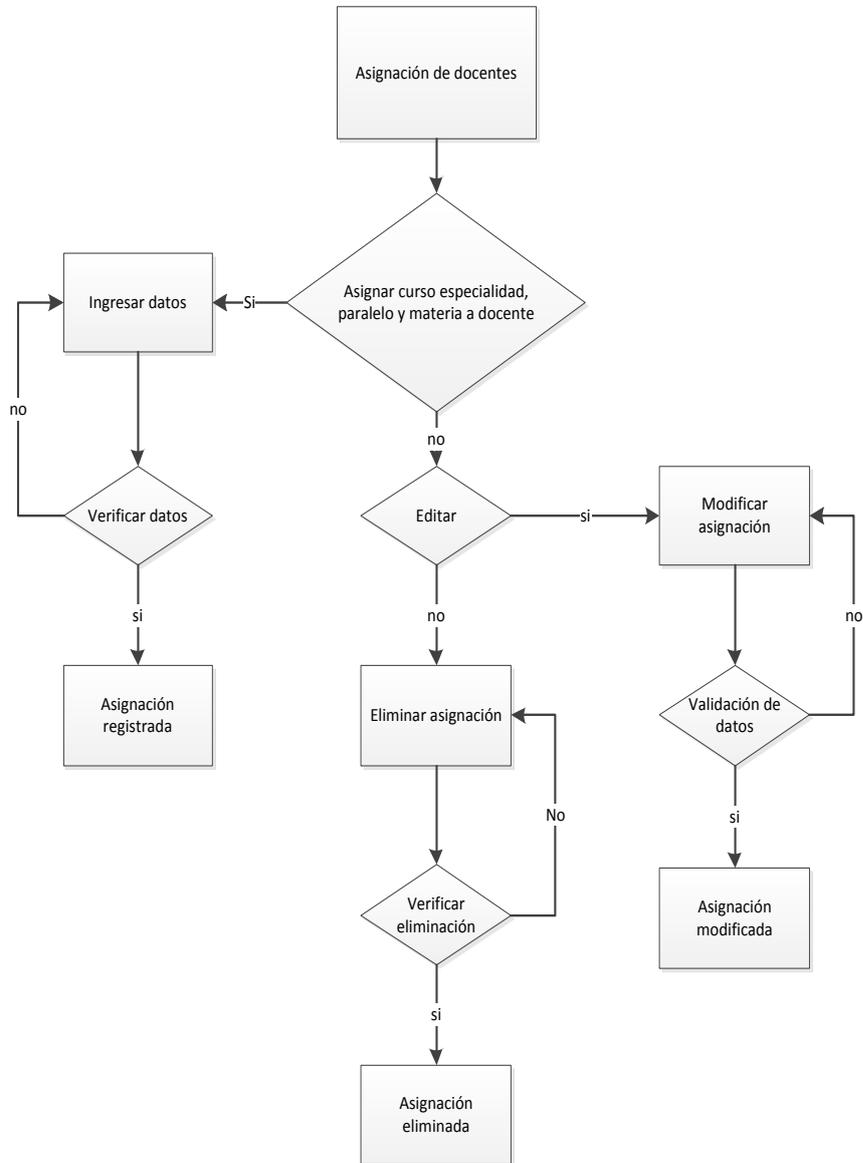


Ilustración 26 Diagrama de Interacción-Asignación de Docentes

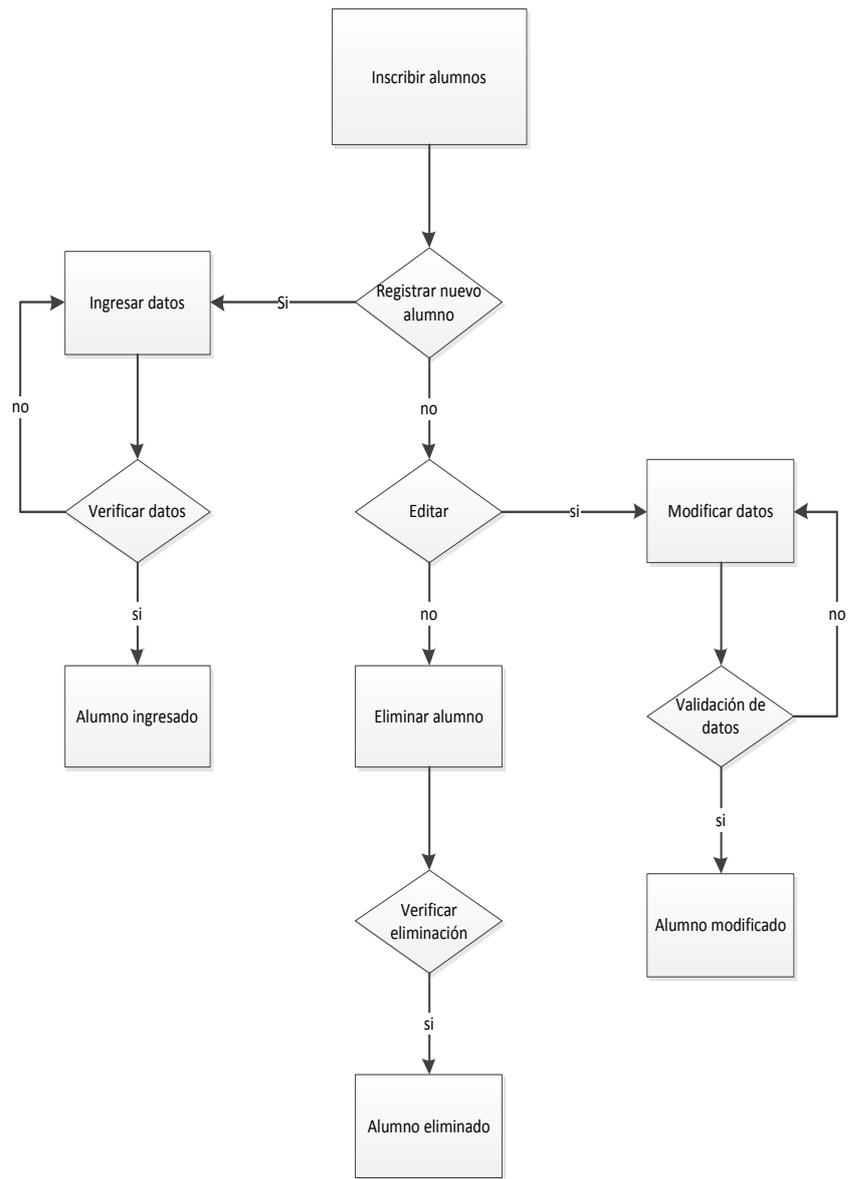


Ilustración 27 Diagrama de Interacción-Gestión de alumnos

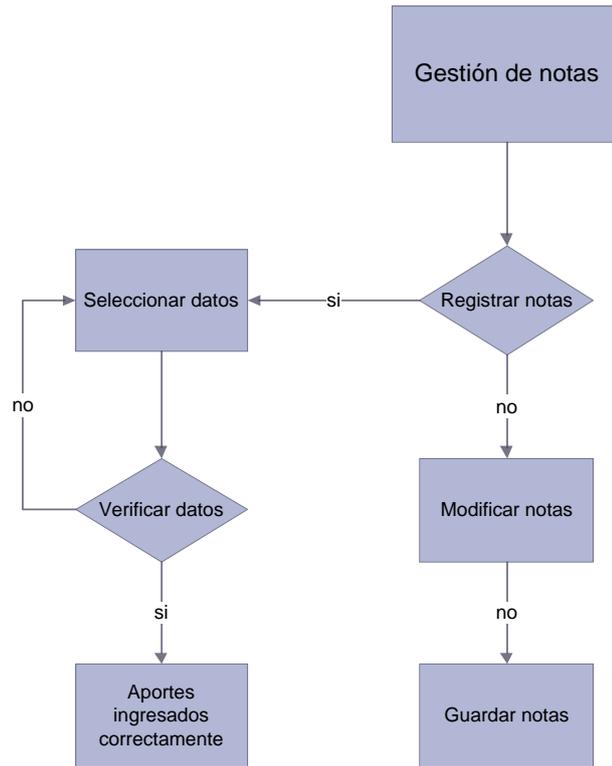


Ilustración 28Diagrama de Interacción-Ingreso de notas

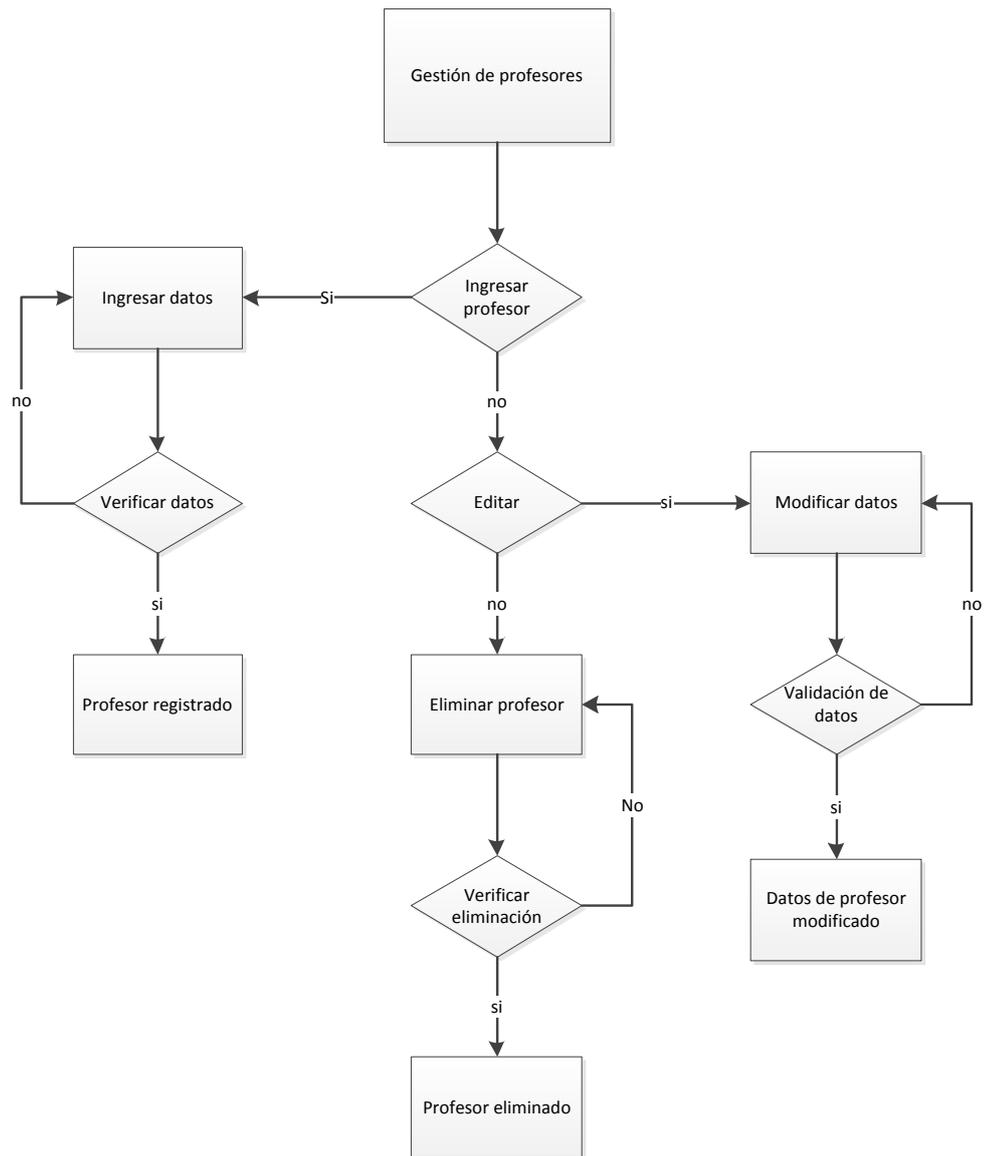


Ilustración 29 Diagrama de Interacción-Docentes

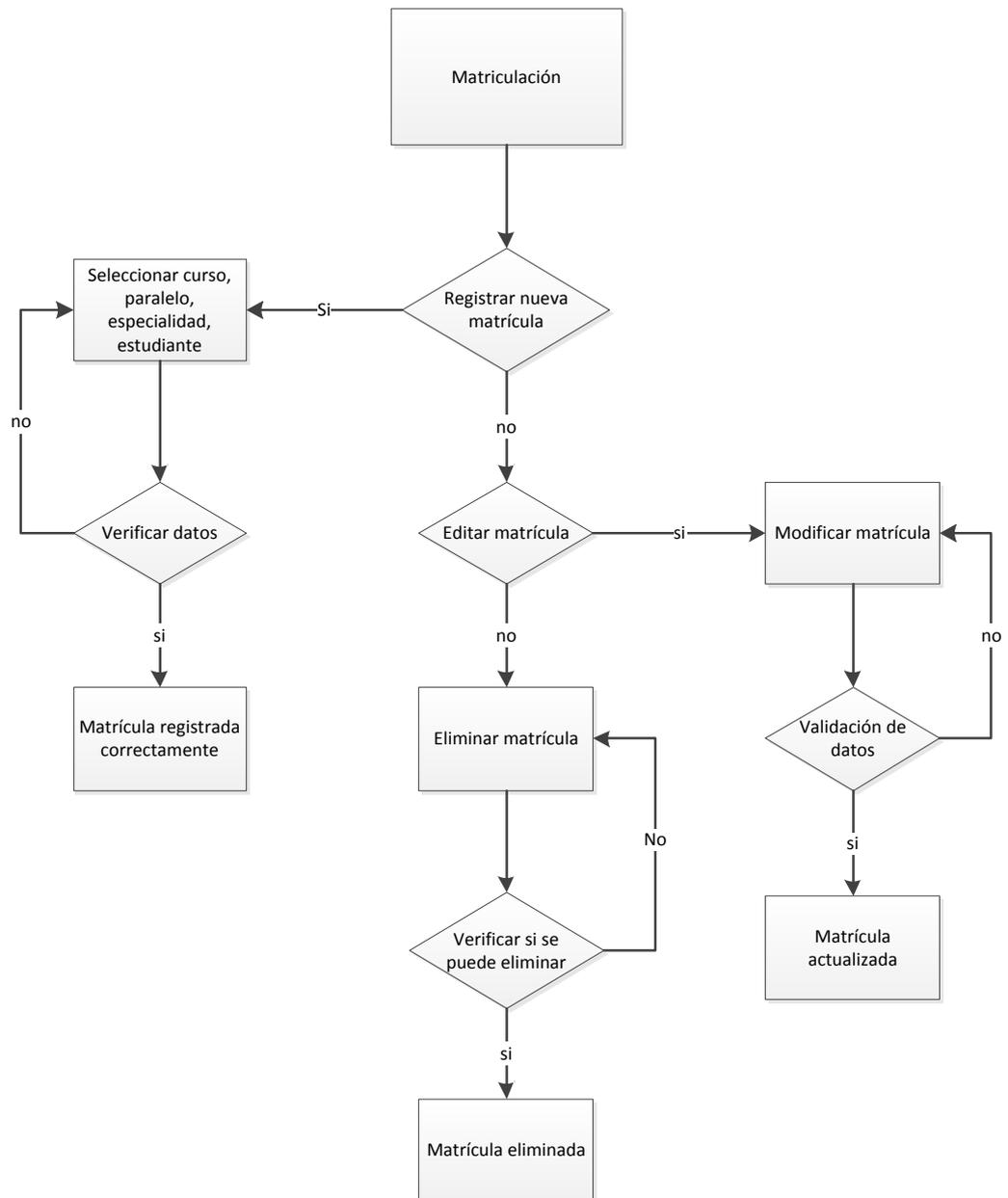


Ilustración 30 Diagrama de Interacción-Matriculas

ANEXO C

MANUAL TÉCNICO

TABLA DE CONTENIDO

INTRODUCCION.....	Pág. 176
OBJETIVO.....	Pág. 176
CONTENIDO.....	Pág. 176

INTRODUCCION:

El presente manual técnico proporcionará la lógica con la que se ha desarrollado la aplicación, la cual se sabe que es propia de cada programador; por lo que se considera necesario ser documentada.

Aclarando que este manual no pretende ser un curso de aprendizaje de cada una de las herramientas empleadas para el desarrollo del sitio, sino documentar su aplicación en el desarrollo del sitio.

OBJETIVO:

Proporcionar una guía del desarrollo de la interfaz y de la puesta en producción del sistema en entorno web SULTANA.

CONTENIDO:

DESARROLLO DE LA INTERFAZ

Desarrollo de página e interfaz de usuario

Software utilizado: AppServ

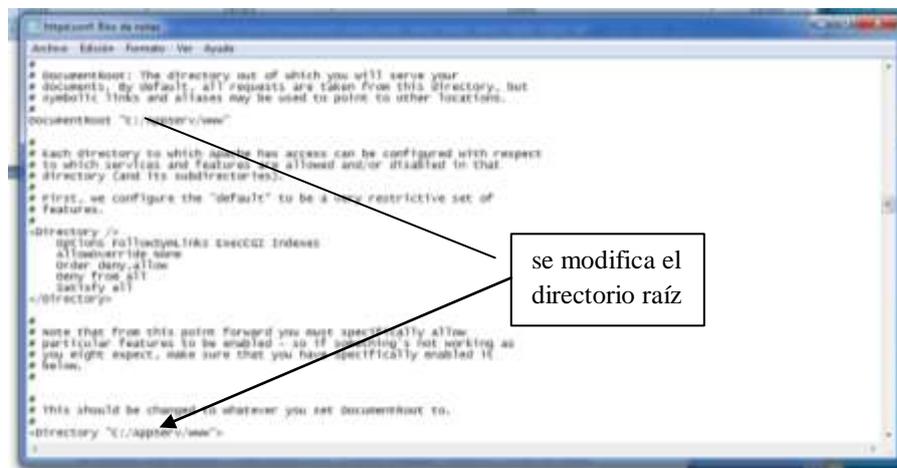
AppServ es una aplicación que permite a los desarrolladores tener un entorno de programación y acceso a datos en una sola instalación y mínima configuración. AppServ esta diseñado para la plataforma Windows, cabe indicar que sirve únicamente para el desarrollo y no es recomendado para servidor definitivo.

los paquetes que nos proporciona esta aplicación son: PHP 5.0, MySQL, Apache

2.0 y de forma opcional el gestor de bases de datos PHPMyAdmin.

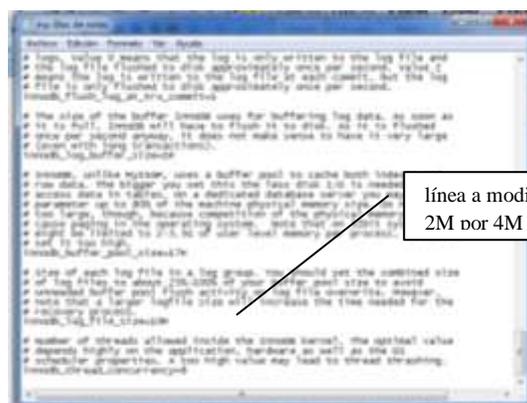
al instalar esta aplicación es necesario introducir un nombre de usuario y contraseña para el acceso al servidor de base de datos.

Luego de terminada la instalación de AppServ se procede a modificar si así lo desea, el directorio raíz donde ira el sitio y a donde accederá el servidor web Apache para buscar el índice de la aplicación, este es el fichero httpd.conf. A continuación la imagen de esta modificación.



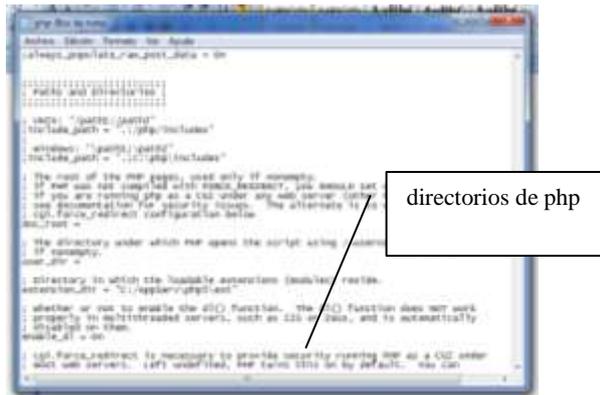
se modifica el directorio raíz

Además, como la base de datos es relacional (innodb) a veces es necesario modificar el tamaño de la memoria de transacción del servidor MySQL en el fichero my.ini del servidor



línea a modificar 2M por 4M

Se pueden modificar los directorios del modulo de php si fuera necesario en el fichero php.ini.



Esta es una aplicación rápida y estable lo que permite al programador agilidad y eficiencia en el desarrollo del proyecto.

Una vez hechas las configuraciones necesarias se procede a implementar la base de datos por medio de phpMyAdmin de acuerdo al modelo ya diseñado.

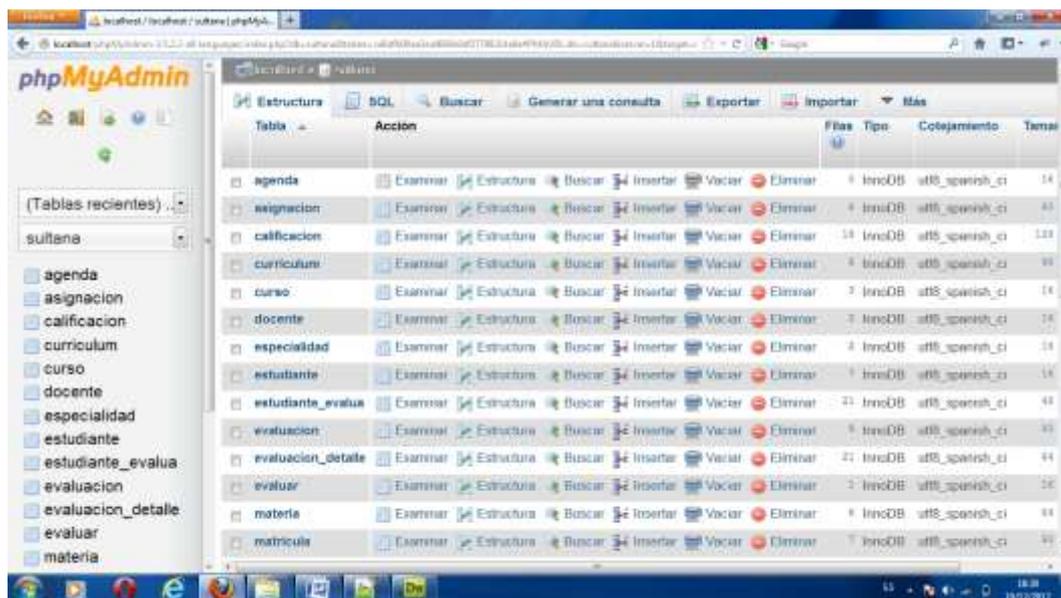


Ilustración 31 Gestión de la base de datos "Sultana"

con estos pasos ya se puede empezar la programación pero sin antes conocer donde y como se organizan los archivos y carpetas por parte de AppServ para la pagina o sistema.

A continuacion se describen a breves rasgos las carpetas y archivos mas importantes utilizados por el proyecto.

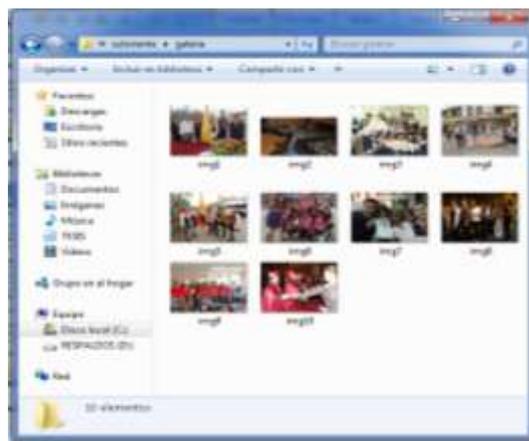


Ilustración 38 carpeta Galería donde se ha guardado las fotos de la galería

Luego de dar una breve descripción de las carpetas y directorios a utilizar se hace una descripción rápida de las diferentes clases y formularios y cadenas de conexión a base de datos.

La pagina principal o index referencia a varias clases que permiten llenar las diferentes tablas como son:

```
window.location="sultana.php";
```

Con este código se accede al formulario "sultana.php"; para acceder al "login" del sistema.



Al ingresar al sistema y de acuerdo a los privilegios que tenga el usuario el sistema pasa al siguiente formulario php llamado "acceso.php" en donde están los iconos de acceso a cada uno de los servicios del sistema. Ya que la propuesta inicial fue la creación del módulo de evaluación, en el presente artículo se detalla únicamente lo referente a EVALUACION"



Como administrador el usuario puede acceder al módulo de evaluación, la lógica del sistema es: se envía un submit conectando automáticamente el formulario de acceso con el de evaluación llamado evaluacion.php por medio del código:

```
<a href="evaluacion.php"><br />Evaluaci&oacute;n</a>.
```

al acceder a evaluación tenemos 4 opciones las cuales se conectan desde evaluacion.php a los diferentes formularios de la siguiente manera.

Para ingreso de unidades:

```
<td class='center'><a href='evaluacion_unidades.php'><img src='../images/unidad.png' width='32' height='32'><br />Unidades</td>
```

Para ingreso de temas:

```
href='evaluacion_temas.php'><img src='../images/tema.png' width='32' height='32'><br />Temas</td>";
```

Para ingresar unidad a evaluar.

```
href='evaluacion_atemas.php'><img src='../images/aevaluar.png' width='32' height='32'><br />A Evaluar</td>
```

Para revisar resultados de la evaluación

```
href='evaluacion_resultados.php'><img src='../images/resultado.png' width='32' height='32'><br />Resultados</td>";
```

La conexión a la base de datos se realiza mediante la clase "conexion.php" y mysql.php, el código de la misma se presenta a continuación:

```
<?php
function Conectarse()
{
echo"
    <title>SULTANA</title>
    <link rel='stylesheet' type='text/css' href='../style/style.css'>
    <link rel='shortcut icon' href='../favicon.ico' >
    <link rel='stylesheet' type='text/css' href='../style/desing.css'>
    ";

    if (!$link=mysql_connect("localhost","sultana","*****"))
    if (!$link=mysql_connect("localhost","root","12345"))
    {
        echo "Error de conexión a la base de datos";
        exit();
    }
    if (!mysql_select_db("sultana",$link))
    {
        echo "Error seleccionando la base de datos";
```

```

        exit();
    }
    return $link;
}

$link=Conectarse();
mysql_close($link); //cierra la conexion

?>

```

Al subir la pagina a la nube es necesario cambiar el nombre de la base de datos y la contraseña de acuerdo a lo dispuesto por el administrador del hosting que aloja la pagina.

Para realizar la programación en si se requirió de clases ya establecidas para aprovechar lo existente en lo posible y el resto como dar formatos, ingreso y selección de datos por programación propia.

Como editor de texto y ya que brinda soporte a php, css y HTML además de la premisa de utilizar software libre se escogió a Notepad++.

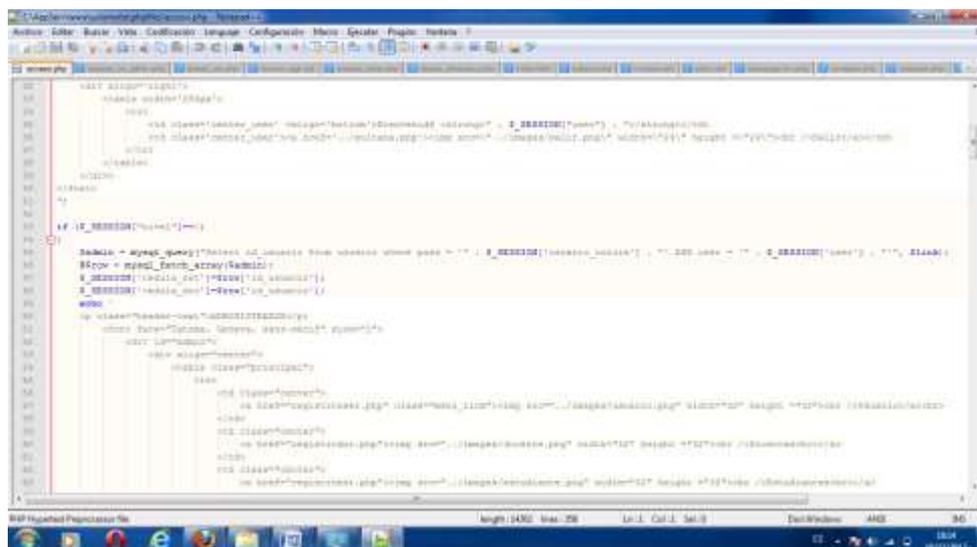


Ilustración 39 Notepad++ donde se realiza la programación en general. en pantalla el código del formulario "acceso.php"

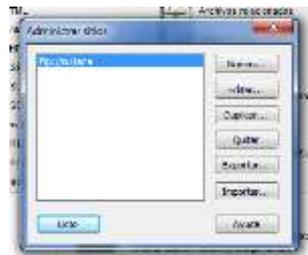
Para visualizar los avances de las iteraciones se emplea firefox de Mozilla de preferencia.

Para subir el sistema al servidor se utiliza Dreamweaver a modo de evaluación.

Pasos para levantar el sitio.

Se necesita que el administrador del hosting provea de un usuario y clave de cliente ftp (protocolo de transferencia de archivos) además de la dirección ftp del servidor.

una vez obtenidos estos datos escogemos en la interfaz de Dreamweaver la pestaña que dice "sitio" escogemos la opción Administrar sitios y presenta la siguiente pantalla:



Aquí creamos una conexión con el servidor con los datos anteriores en la pestaña que dice nuevo.



una vez hecho esto y probado la conexión ya podemos subir los archivos de la pagina y nos queda como en la siguiente ilustración:

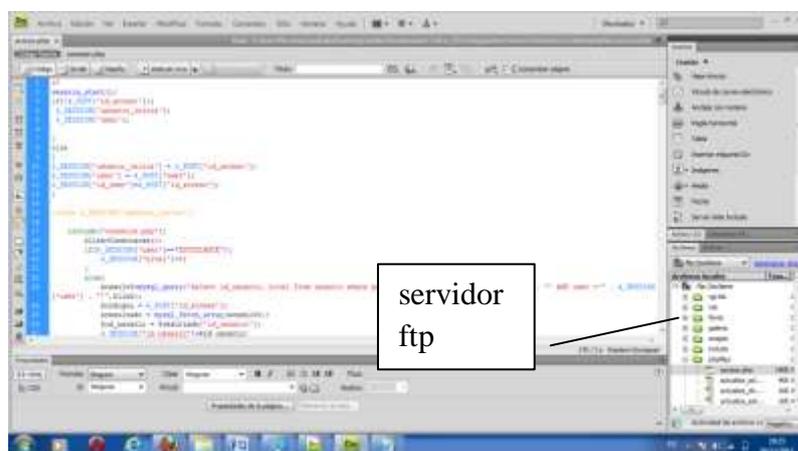


Ilustración 40 Subiendo el sitio y administrando por medio de servidor ftp

Una vez que el sistema está en producción podremos seguir con la fase de mantenimiento por medio de el cliente ftp creado en Dreamweaver.

La desventaja más significativa en la elaboración de este proyecto es la utilización de Notepad para programar y no una herramienta de programación como por ejemplo "easyphp" o el mismo "Dreamweaver".

MANUAL DE USUARIO

1	INDICE	186
2	ACERCA DEL PROGRAMA	186
2.1	INFORMACIÓN	186
2.2	REQUERIMIENTOS	187
3	USO Y FORMAS DE OPERACIÓN	187
3.1	PANTALLA PRINCIPAL	187
3.2	ADMINISTRADOR	188
3.2.1	Usuario	188
3.2.2	Docente	189
3.2.3	Estudiante	189
3.2.4	Materias	190
3.2.5	Materias a docentes	190
3.2.6	Matriculas	191
3.2.7	Ingreso de Unidades	191
3.2.8	Ingreso de Temas por Unidades	192
3.3	USUARIO ESTUDIANTE	192
3.3.1	Usuario Estudiante actualizar datos	193
3.3.2	Consultar Calificaciones	193
3.3.3	Realizar Evaluaciones	194
3.3.4	Resultados de las Evaluaciones	194
3.3.5	Reportes de los Resultados de las Evaluaciones	195

2 ACERCA DEL PROGRAMA

2.1 INFORMACIÓN

Este sistema de Evaluación Curricular es un una aplicación web que permite realizar la revisión de los contenidos de las materias dictadas en el tercer año de Bachillerato de esta institución. Al inicio del año el Ministerio de Educación envía las unidades de las asignaturas y sus contenidos, entonces el administrador del

sistema o docentes ingresaran los mismos, de los cuales deberán planificar para poder cumplir con todas las unidades en el transcurso del año.

Al finalizar el ciclo o cuando los directivos lo requieran podrán pedir reportes sobre las materias revisadas y tomar sus debidos correctivos a tiempo.

2.2 REQUERIMIENTOS

La página Web se encuentra ya disponible para las siguientes plataformas

- Hardware – PC compatible x86 (i386, i486, Pentium I, II, III, IV, K6, K6-2, Duron, Athlon, etc.)
 - 10 Mb libres en el disco rígido
- Software – Windows 9x, Me, NT, 2000, Vista. Windows 7
 - Linux (kernel > 2.0) con librería gráfica QT

3.- USO Y FORMAS DE OPERACIÓN

3.1 Pantalla Principal.- La pantalla principal de la página web tiene como fin

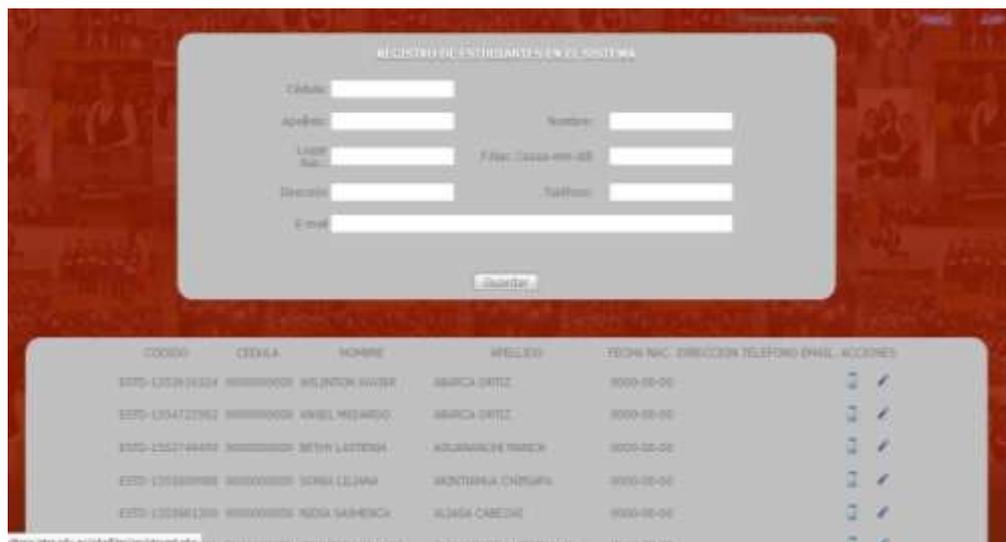


El organigrama de trabajo, historia de la institución, misión, visión, En personal Docente, las especialidades existentes, la malla , los servicios que ofrece el colegio, galería de fotos de eventos institucionales que se han dado durante la trayectoria institucional y contactos en la cual consta dirección, teléfonos, celular, correo electrónico para consultas, separar citas y receptar sugerencias. Además los usuarios de la página como lo son estudiantes, docentes y personal administrativo ingresaran su nombre de usuario y clave en el lugar donde dice plataforma.

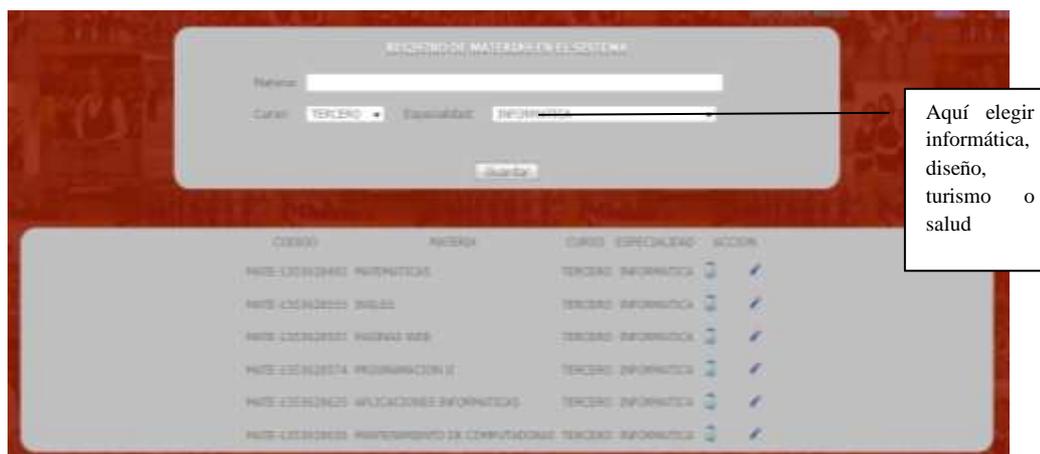
3.2 Administrador.- Después de haber ingresado su nombre de usuario y clave personal el usuario tiene acceso a la pantalla de Administrador, en donde el podrá visualizar los usuarios del sistema así como podrá borrarlos y modificar sus datos. De este modo también podrá hacer el ingreso de docentes, estudiantes, materias, cursos, paralelos, asignaturas, matriculas, notas, evaluaciones y poder imprimir reportes.



3.2.1 Usuario.- Después de haber ingresado su nombre de usuario y clave personal el usuario tiene acceso a la pantalla de Administrador, en donde el podrá visualizar los usuarios del sistema así como podrá borrarlos y modificar sus datos. De este modo también podrá hacer el ingreso de docentes, estudiantes, materias, cursos, paralelos, asignaturas, matriculas, notas, evaluaciones y poder imprimir reportes.



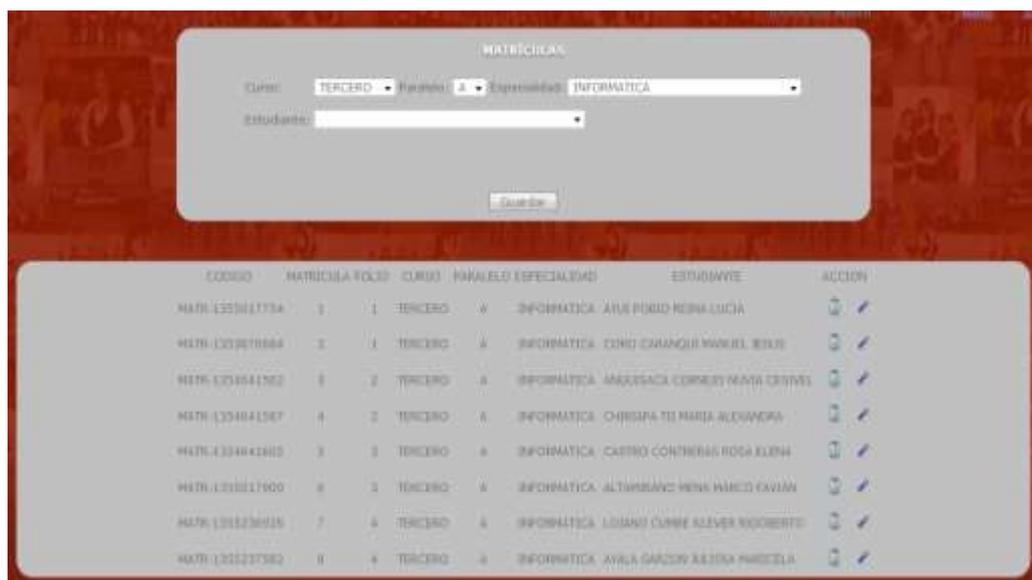
3.2.4 Materias.- las asignaturas serán ingresadas mediante el administrador del sistema según las especialidades que existan, el curso y las materias de cada especialidad.



3.2.5 Materias a Docentes.- las asignaturas serán asignadas mediante el administrador del sistema según las especialidades que existan, el curso y las materias de cada especialidad. Cada docente cumplirá con un total máximo de 30 horas semanales.



3.2.6 Matriculas.- Una vez que el estudiante ha sido ingresado en el sistema con todos sus datos personales, en el periodo de inscripciones o estudiantes existentes en el plantel, se les asignara su respectivo curso, paralelo y especialidad, según el cupo existente en cada área.



3.2.7 Ingreso de Unidades.- Una vez las asignaturas han sido ingresado en el sistema con todas sus especificaciones en cada periodo académico se deben modificar o revisar las materias y contenidos, y una vez hecho esto, se buscara el curso, especialidad, materias y docente. Posteriormente se pone la unidad que se desea ingresar acompañado de su descripción y objetivos planteados.

INGRESO DE UNIDADES

Curso: **TEC2302** Especificidad: **INFORMATICA**

Materia: **PAGINAS WEB** Docente: **MAIRELA**

Unidad: **1**

Descripción: **HTML**

Objetivo: **ESTUDIAR EL LENGUAJE WEB HTML**

Guardar

CODIGO	UNIDAD	DESCRIPCION	OBJETIVO	ACCION
UNID-1254497260	1	HTML	ESTUDIAR EL LENGUAJE WEB HTML	
UNID-1254497322	1	IMÁGENES Y SUS FORMATOS	ESTUDIAR LOS FORMATOS GIF, JPEG Y MANEJO DE IMÁGENES	
UNID-1254497395	1	TABLAS, FRAMES, CAMPOS FORMULARIOS Y HOJAS DE ESTILO	APRENDER EL TRATAMIENTO DE TABLAS, FRAMES, CAMPOS FORMULARIOS Y HOJAS DE ESTILO	
UNID-1254497472	4	CSS, VALIDACIÓN	COMO SUBIR UNA PAGINA A LA WEB, REALIZAR UN PROYECTO PRACTICO	

3.2.8 Ingreso de Temas por Unidades.- Los temas a ingresar deberán contar con la aprobación del Rector y de la Junta Directiva.

INGRESO DE TEMAS POR UNIDAD

Curso: **TEC2302**
Especificidad: **INFORMATICA**
Materia: **PAGINAS WEB**
Docente: **MAIRELA**

Seleccionar la Unidad: **1** Ver: Todos los Temas

Guardar Ver Unidad

UNIDAD	TEMA	ACCIONES
1	1. El Lenguaje HTML	
1	2. El Primer Documento HTML	
1	3. Características Generales de HTML	
1	4. Marcas y Atributos	
1	5. Normas Fundamentales	

3.3 Usuario Estudiante.- Después de haber ingresado su nombre de usuario y clave personal el estudiante tiene acceso a la pantalla de estudiante, en donde el

podrá visualizar sus calificaciones, actualizar sus datos en caso de cambio de domicilio o de correo electrónico, ver sus tareas, notificaciones, agenda telefónica y por ultima la tarea mas importante que es la de evaluar el desempeño del docente al impartir los temas de las unidades.



3.3.1 Usuario Estudiante actualizar datos.- Al instante de ingresar los datos el administrador pudo cometer algún error, permitiéndole al estudiante modificarlo sin ningún problema de la siguiente manera.



3.3.2 Consultar Calificaciones.- El estudiante antes de poder ver sus calificaciones o reportes, deberá primeramente evaluar los contenidos de las materias, después podrá ver sus calificaciones y posteriormente poder realizar sus respectivos correctivos con el docente o también servirá de información para sus padres.

CONSULTA DE NOTAS	
CURSO:	TERCERO
ASIGNATURA:	INFORMATICA
ESTUDIANTE:	CORO CARANQUI MANUEL JESUS
MATEMA	1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0 10.0
INGLES	1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0 10.0
MATEMÁTICAS	1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0 10.0

3.3.3 Realizar Evaluaciones.- El estudiante tiene la posibilidad de evaluar los contenidos revisados por los docentes, cada evaluación se deberá realizar mediante la unidad que se desea revisar, en la parte izquierda de la pantalla se pueden visualizar los temas que pertenecen a cada unidad y en la parte derecha se puede evaluar según los temas estudiados.



3.3.4 Resultados de las Evaluaciones.- El administrador mediante orden de sus superiores podrá entrar a esta pantalla de resultados los cuales se los puede visualizar de manera gráfica el porcentaje de los temas revisados por unidad y según esto dar un reporte del cumplimiento del docente y de los contenidos planteados en la planificación.



3.3.5 Reportes de los Resultados de las Evaluaciones.- En junta directiva se pide al administrador por impreso y firmado los reportes dados por el sistema y según esto poder tomar decisiones para poder cumplir con la planificación anual curricular.

**COLEGIO PARTICULAR A DISTANCIA
SULTANA DEL ORIENTE**

Curso: TERCERO
Especialidad: INFORMATICA
Materia: PAGINAS WEB
Docente: JARAMILLO MARIELA

Estudiantes Evaluados: 1
Cumplimiento según Estudiantes: 75.00%
Cumplimiento Items Individual: 75.00%

Num	TEMA	VISTO.	NO VISTO.	Porcentaje.
1	1. El Lenguaje HTML	1	0	100.00%
2	2. Mi Primer Documento HTML	1	0	100.00%
3	3. Características Generales de HTML	1	0	100.00%
4	4. Marcas y Atributos	0	1	0.00%
5	5. Normas Fundamentales	0	1	0.00%
6	6. Estructura de un Documento HTML	0	1	0.00%
7	7. Comandos Básicos de HTML	1	0	100.00%
8	8. Definición de Páginas: P, Br.	1	0	100.00%
9	9. Estilos de Texto	1	0	100.00%
10	10. Listas OL, UL, DL	1	0	100.00%
11	11. Texto: Centrado, Líneas, Comentarios	1	0	100.00%
12	12. Enlaces a otras zonas de la página	1	0	100.00%