



UNIVERSIDAD TECNOLÓGICA EQUINOCCIAL

FACULTAD DE CIENCIAS DE LA INGENIERÍA CARRERA INFORMÁTICA Y CIENCIAS DE LA COMPUTACIÓN

“IMPLEMENTACIÓN DE UN SISTEMA LMS (MOODLE) EN LA SECCIÓN NOCTURNA TÉCNICA DEL COLEGIO INTISANA DE QUITO”

**TRABAJO PREVIO LA OBTENCIÓN DEL TÍTULO DE INGENIERO
EN INFORMÁTICA Y CIENCIAS DE LA COMPUTACIÓN**

CISNEROS TELLO GILBERT ADRIAN

DIRECTOR: ING. VICTOR HUGO GALVEZ CAZA

Quito, Febrero 2013

© Universidad Tecnológica Equinoccial. 2013

Reservados todos los derechos de reproducción

DECLARACIÓN

Yo **GILBERT ADRIAN CISNEROS TELLO**, declaro que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

La Universidad Tecnológica Equinoccial puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.

Gilbert Adrián Cisneros Tello

C.I. 1716568892

CERTIFICACIÓN

Certifico que el presente trabajo que lleva por título **“IMPLEMENTACIÓN DE UN SISTEMA LMS (MOODLE) EN LA SECCIÓN NOCTURNA TÉCNICA DEL COLEGIO INTISANA DE QUITO”**, que, para aspirar al título de **Ingeniero en Informática y Ciencias de la Computación** fue desarrollado por **Gilbert Adrián Cisneros Tello**, bajo mi dirección y supervisión, en la Facultad de Ciencias de la Ingeniería; y cumple con las condiciones requeridas por el reglamento de Trabajos de Titulación artículos 18 y 25.

Ing. Víctor Hugo Gálvez Gaza

DIRECTOR DEL TRABAJO

C.I. 1707796536

Quito, 23 de febrero de 2012

Estimado

Sr. Adrián Cisneros Tello

Remitimos esta carta como contestación a su petición, afirmando que el desarrollo e implementación de la tesis, "IMPLEMENTACIÓN DE UN SISTEMA LMS (MOODLE) AL COLEGIO INTISANA DE QUITO" cumple y supera satisfactoriamente todas las expectativas, la implantación del sistema, ayuda de gran manera al proceso de estudio de la sección nocturna de la institución así como un respaldo para la sección diurna.

Agradeciendo el trabajo brindado para la institución en la aplicación de su tesis para el apoyo a estudiantes de pocos recursos económicos, demás estudiantes y profesores.

Usted puede hacer uso de este documento a su conveniencia.

Atentamente

COLEGIO INTISANA

DEDICATORIA

Dedico este trabajo de tesis a en primero a Dios, a mi familia, en especial a mi madre y mi hermano, personas quienes siempre han estado a mi lado, apoyándome y dándome aliento para cumplir con mis metas y hacer mis sueños realidad.

A mi madre lo dedico le dedico mi primer título superior como un comienzo de estudios en mi vida, un puente cruzado para continuar adelante en mi vida estudiantil.

A mi hermano como ejemplo de que toda meta es un paso para cumplir sueños, hacerlos realidad y seguir adelante, siendo hombres integrales, buenos en todo aspecto de la vida.

De Igual Manera dedico este plan de titulación a todos los profesores con los que tuve el honor de estar en mis años estudiantiles, ya que en base a sus enseñanzas he logrado concluir esta etapa estudiantil.

AGRADECIMIENTO

Agradezco a la Universidad Tecnológica Equinoccial, mi universidad la cual me ha dado las oportunidades de trabajo desde el primero a mis 19 años, hasta el tiempo presente, Gracias por permitirme crecer en el ámbito estudiantil, laboral y Personal.

A las personas que hicieron posibles esta tesis, mi director de tesis y gran profesor, amigo de quien he aprendido mucho para mi vida personal, Ing. Victor Hugo Galvez; al Ing. Juan Bravo quien con su ayuda contacto al colegio Intisana para poder realizar esta tesis con vinculación a la sociedad.

Al Colegio Intisana por permitirme hacer uso de sus aulas tiempo de estudiantes y profesores, a quien en manera particular agradezco de gran manera al Ing. Robert Montero, quien ayudo de gran manera al proceso para que la implementación sea un éxito.

Finalmente más que un agradecimiento es un gracias con un gran abrazo a mis compañeros y amigos que estuvieron en mi vida estudiantil en especial a Karina Bajaña, Jorge Jaramillo y Juan Carlos Lafuente, gracias por estar en cada momento.

ÍNDICE DE CONTENIDOS

	PÁGINA
Resumen	Vii
Abstract	Viii
1 INTRODUCCIÓN.....	1
2 MARCO TEÓRICO.....	5
2.1 PLATAFORMA VIRTUAL	5
2.1.1 LMS	5
2.1.2 PROCESO ENSEÑANZA APRENDIZAJE	6
2.1.3 EDUCACIÓN VIRTUAL.....	7
2.2 APLICACIONES WEB.....	7
2.2.1 MOODLE – Software educativo en Línea	8
2.2.2 HERRAMIENTAS Y PLATAFORMAS DE DESARROLLO.	9
2.2.2.1 Rational Rose	9
2.2.2.2 Base de Datos.....	10
2.3 METODOLOGÍAS.....	11
2.3.1 METODOLOGÍA DE INVESTIGACIÓN	11
2.3.1.1 MÉTODO DEDUCTIVO	11
2.3.2 METODOLOGÍA INFORMÁTICA.....	11
2.3.3 METODOLOGÍA DE DESARROLLO.....	12
2.3.3.1 Visión general de la programación extrema	18
2.3.3.2 Análisis y diseño en programación extrema	20

2.3.3.3	Historias del usuario.....	20
2.3.3.4	Modelos de negocio	21
2.3.3.5	Interacción entre las cuatro variables de la gestión de proyecto	22
2.3.3.6	Actores y funciones en programación extrema.....	23
2.3.3.7	Fases de la programación extrema.....	25
2.3.3.8	Comparativa con otras metodologías tradicionales	32
2.3.3.9	Beneficios y desventajas de la programación extrema.....	34
3	METODOLOGÍA	36
4	ANÁLISIS Y DISCUSIÓN DE RESULTADOS	39
4.1	FASE INICIAL – ANÁLISIS.....	39
4.1.1	INTRODUCCIÓN	39
4.1.1.1	Propósito	39
4.1.1.2	Ámbito del Sistema	39
4.1.1.3	Definiciones, acrónimos y abreviaturas	40
4.1.1.4	Referencias	40
4.1.2	DESCRIPCIÓN GENERAL	40
4.1.2.1	Perspectiva del producto.....	40
4.1.2.2	Funciones del producto	41
4.1.2.3	Características de los Usuarios	42
4.1.2.4	Restricciones.....	42
4.1.3	REQUISITOS ESPECÍFICOS.....	43
4.1.3.1	Requisitos de interfaces	43

4.1.3.2	Requisitos no Funcionales	45
4.1.3.3	Requisitos Funcionales	46
4.2	FASE DE ELABORACIÓN – DISEÑO	52
4.2.1	PLANIFICACIÓN.....	53
4.2.1.1	FACTIBILIDAD	54
4.2.2	DISEÑO.....	64
4.2.2.1	DISEÑO DE INTERFACES DE USUARIO	64
4.2.2.2	DISEÑO DE LA BASE DE DATOS.....	67
4.2.3	CODIFICACIÓN	69
4.2.3.1	CREACIÓN DE SECCIONES DEL COLEGIO INTISANA.....	69
4.2.3.2	CREACIÓN DE USUARIOS	72
4.2.3.3	IMPLANTACIÓN	74
4.2.4	PRUEBAS	75
4.2.4.1	RESULTADOS.....	77
4.2.4.2	DISCUSIONES.....	90
5	CONCLUSIONES Y RECOMENDACIONES	92
5.1	CONCLUSIONES	92
5.2	RECOMENDACIONES.....	93
6	BIBLIOGRAFÍA.....	96
7	ANEXOS	98

ÍNDICE DE TABLAS

	PÁGINA
TABLA 1. INTERACCIÓN ENTRE LAS CUATRO VARIABLES DE LA GESTIÓN DE PROYECTO.....	22
TABLA 2. ESPECIFICACIÓN DE CU: ACCESO Y SALIDA DEL SISTEMA	46
TABLA 3. DESCRIPCIÓN DE CU DEL ADMINISTRADOR.....	48
TABLA 4. DESCRIPCIÓN DE CASOS DE USO DEL PROFESOR.....	49
TABLA 5. DESCRIPCIÓN DE CASOS DE USO DEL ALUMNO	50
TABLA 6. DESCRIPCIÓN CASO DE USO INTERACCIÓN ENTRE PROFESOR Y ALUMNO	51
TABLA 7 PREGUNTAS PARA CONFIRMACIÓN DE METODOLOGÍA A UTILIZAR.....	52
TABLA 8. ANÁLISIS FODA	55
TABLA 9. FACTIBILIDAD TÉCNICA DE SOFTWARE	59
TABLA 10. FACTIBILIDAD TÉCNICA DE HARDWARE	60
TABLA 11. ANÁLISIS DE COSTOS	61
TABLA 12. URLS USADAS PARA PRUEBAS DE RIGOR.....	77
TABLA 13. CONFIGURACIÓN PARA LA PRUEBA TIME.....	78
TABLA 14. RESULTADOS POR USUARIO EN PRUEBA TIME	79
TABLA 15. RESULTADO POR URL EN PRUEBA TIME.....	80
TABLA 16. CONFIGURACIÓN PARA PRUEBAS RAMP	84
TABLA 17. RESULTADOS POR USUARIOS EN PRUEBA RAMP	85
TABLA 18. RESULTADO POR URL EN PRUEBA RAMP.....	86
TABLA 19. RESULTADOS DE IMPLANTACIÓN DEL SISTEMA.....	90

ÍNDICE DE FIGURAS

	PÁGINA
FIGURA 1. PROCESO DE ENSEÑANZA APRENDIZAJE	6
FIGURA 2. PILARES DE LA PROGRAMACIÓN ORIENTADA OBJETOS	12
FIGURA 3. HISTORIAS DEL USUARIO	21
FIGURA 4. FASES DE LA METODOLOGÍA DE XP	25
FIGURA 5. UN DÍA DE TRABAJO CON XP	32
FIGURA 6 COMPARATIVO DE XP CON OTRAS METODOLOGÍAS	33
FIGURA 7. ACCESO Y SALIDA DEL SISTEMA	46
FIGURA 8. CASOS DE USO DEL ADMINISTRADOR	47
FIGURA 9. CASOS DE USO DEL PROFESOR	48
FIGURA 10. CASOS DE USO DEL ALUMNO	49
FIGURA 11. CASO DE USO INTERACCIÓN ENTRE PROFESOR Y ALUMNO	50
FIGURA 12. FASES METODOLOGÍA XP	53
FIGURA 13. PANTALLA BASE ADMINISTRADOR	65
FIGURA 14. PANTALLA BASE ESTUDIANTE	66
FIGURA 15. PANTALLA CURSOS PROFESOR	66
FIGURA 16. PANTALLA BASE ADMINISTRACIÓN CURSO PROFESOR	67
FIGURA 17. MODELO ENTIDAD RELACIÓN DE LA BDD DE MOODLE.	68
FIGURA 18. PRIMER PASO PARA AGREGAR NUEVO CURSO	69
FIGURA 19. SEGUNDO PASO PARA AGREGAR NUEVO CURSO	70
FIGURA 20. TERCER PASO PARA CREACIÓN DE CURSOS	71
FIGURA 21. CUARTO PASO PARA CREACIÓN DE CURSOS	71
FIGURA 22. CREACIÓN DE ARCHIVO CSV PARA CARGA MASIVA	72
FIGURA 23. PANTALLA PARA SUBIR USUARIOS MASIVAMENTE	74
FIGURA 24. PANTALLA PRINCIPAL DE INGRESO	75

FIGURA 25. INGRESO DE USUARIOS EN SISTEMA PRUEBAS SATISFACTORIAS	76
FIGURA 26. TIEMPO DE RESPUESTA EN PRUEBA TIME	80
FIGURA 27. USO DE ANCHO DE BANDA EN PRUEBA TIME	81
FIGURA 28. USO DEL CLIENTE EN PRUEBA TIME	81
FIGURA 29. PROMEDIO DE ESPERA EN RESPUESTA A REQUERIMIENTO EN PRUEBA TIME	82
FIGURA 30. PRUEBA DE ERRORES POR USUARIOS EN CADA URL EN PRUEBA TIME	82
FIGURA 31. TIEMPO DE RESPUESTA EN PRUEBA RAMP	87
FIGURA 32. USO DE ANCHO DE BANDA EN PRUEBA RAMP	87
FIGURA 33. USO DEL CLIENTE EN PRUEBA RAMP	88
FIGURA 34. PROMEDIO DE ESPERA EN RESPUESTA A REQUERIMIENTO EN PRUEBA RAMP	88
FIGURA 35. PRUEBA DE ERRORES POR USUARIOS EN CADA URL EN PRUEBA RAMP	89
FIGURA 36 DIAGRAMA CAMBIO DE VERSIÓN	98

ÍNDICE DE ANEXOS

	PÁGINA
ANEXO 1.	
Diagrama Cambio de Versión.....	98
ANEXO 2.	
Manual del Usuario.....	99
ANEXO 3.	
Esquema de la Base de Datos.....	100
ANEXO 4.	
Firmas capacitación de estudiantes.....	110

RESUMEN

La implantación del LMS MOODLE en la sección técnica nocturna del colegio Intisana de Quito es un proyecto de vinculación Social, el cual se lo realizó ya que se dedujo que el tiempo presencial de los estudiantes en las aulas no es suficiente para afianzar sus conocimientos, por tanto era necesaria una plataforma virtual en la cual tanto profesores como estudiantes tienen acceso desde cualquier sitio y a cualquier hora, y mediante la cual profesores envían trabajos, revisan deberes y demás acciones educativas, de igual manera los estudiantes realizan preguntas tanto a sus compañeros como a su profesor haciendo uso del chat o de la mensajería que ésta aplicación ofrece, moodle tiene muchos otros módulos los cuales sirven para que el tiempo presencial de los profesores y estudiantes sea ocupado de una mejor manera. En el sistema también se realizó la creación de usuarios, y perfiles (Administrador, Profesor, Estudiante), con esto se generó seguridad en él, y en consecuencia un sistema seguro que demanda poco mantenimiento, además el sistema tiene integrado los módulos de chat, blog, foros, entre otros, los cuales permiten una comunicación en tiempo real entre los estudiantes y profesores. Una vez generada la Información con la que trabajara el sistema se implanto la estructura de secciones y cursos con la que cuenta el colegio, de esta manera los profesores y estudiantes ya pudieron hacer uso del mismo, cuando fueron alcanzados los objetivos principales y después de poner en marcha el sistema con los primeros cursos, esto se replicó inclusive a la sección diurna lo que demostró la eficiencia del sistema en su uso e implantación.

ABSTRACT

The implementation of the LMS MOODLE in the night technical section of Intisana high school in Quito is a project of social bonding, which was performed since the time it appeared that attendance of students in classrooms is not enough to boost their knowledge, therefore an virtual platform was needed in which both teachers and students have access from anywhere, anytime, and through which teachers submit jobs, review homework and other educational activities, just as the students take both their peers questions as to their teacher using the chat or messaging application it offers, Moodle has many other modules that serve to make the time attendance of teachers and students is engaged in a better way. Into the system was also created users and profiles (administrator, teacher, student), whit this was create a secure system, and consequently a system that demands very little maintenance, also the system has integrated modules like chat, blog, forums, among others, which allow real-time communication between students and teachers. Once generated the information to work with the system structure was implemented with the sections and courses which counts the school, so teachers and students and were able to use it, when the main objectives were achieved and after launching the system with the first courses, the process was replied to the daily section which showed the efficiency of the system in use and implementation.

INTRODUCCIÓN

1 INTRODUCCIÓN

El Colegio Intisana se localiza en Ecuador, en el noroccidente de la ciudad de Quito, provincia Pichincha, es una Institución que tiene el carácter de Unidad Educativa Experimental y cuenta con autonomía pedagógica, administrativa, organizativa y técnica, consta de dos secciones (Diurna y Nocturna), conformado con 107 profesores y mas de 700 alumnos. A más de una infraestructura moderna, amplias áreas verdes y en una ubicación estratégica para llevar a cabo las actividades académicas durante sus jornadas.

La Sección Nocturna sección en la cual se implanta la solución, comenzó a partir de 1970 con un curso de preparación y nivelación para todos los alumnos del sector de Cochapamba, el cual duró 1 mes y al que asistieron unos 20 alumnos de 14 a 22 años. El 8 de Diciembre de 1973, comenzó oficialmente ésta Sección con treinta alumnos fundadores, de los cuales seis se graduaron de bachilleres.

Económicamente, la Sección Nocturna Técnica depende de la Sección Diurna. Aunque los alumnos de la Nocturna cancelan mensualmente la cantidad de 5,00 dólares americanos por concepto de pensión y al inicio del año deben cancelar 3,00 dólares americanos por concepto de matrícula; el dinero recaudado no alcanza para cubrir ni siquiera las necesidades básicas de la Sección, cabe añadir además que existen alumnos becados. La infraestructura física de la Sección Nocturna tiene aproximadamente 2000 metros cuadrados de construcción distribuidos en cuatro plantas, cuenta actualmente con 2 laboratorios de Informática equipados con 35 computadoras, los cuales se encuentran equipados con computadoras de última generación para estudiantes. En la segunda planta está el Laboratorio de Electricidad Básica, en la tercera planta el Laboratorio de Motores y en la cuarta planta el Laboratorio de Electrónica.

Tomando en cuenta estos datos como base de la necesidad de crear una herramienta para el uso de la sección nocturno que a pesar de tener bajos recursos, tiene la tecnología necesaria para la utilización de un aula virtual que ayuda de gran manera al proceso de estudio y no representa un costo alto para la institución, el presente trabajo de tesis está orientado a mejorar la interacción en el proceso de aprendizaje entre el profesor y sus alumnos, haciendo uso de herramientas informáticas, para los institutos, colegios, y universidades, en la actualidad se opta por la utilización de plataformas virtuales las cuales tiene varias opciones de utilización, entre estas están el subir y bajar archivos como deberes, trabajos o consultas, además cuentan con opciones de foros y chat entre estudiantes y profesores, la opción de poder realizar pruebas en el sistema y la auto evaluación de las mismas es una de las características que ayuda de gran manera al proceso. Tomando en cuenta todas estas opciones para estudiantes y profesores en estos tiempos donde la utilización del internet es cotidiana, se decide realizar la implementación de la plataforma MOODLE para la sección nocturna del colegio Intisana de Quito.

Al ser el colegio Intisana una unidad educativa experimental decidió poner en práctica el proyecto implantación de un sistema LMS MOODLE, el cual brinda una educación innovadora, diferente de la educación tradicional porque utiliza herramientas informáticas como correo electrónico, páginas web, foros de discusión y mensajería instantánea. En los últimos años este tipo de modalidad ha aumentado su utilización, pues ya son cientos de universidades, instituciones y centros educativos que utilizan Internet para ofrecer al público la oportunidad de cursar ya sea una carrera o un posgrado completo, con esto se cumple el cumplimiento del objetivo general el cual es:

- Implementar el sistema MOODLE en la sección nocturna técnica del Colegio Intisana.

MOODLE también es un gestor de cursos en línea, estos son entornos creados específicamente para dicho fin; contienen herramientas que apoyan el aprendizaje del alumno, siendo también un programa que desempeña el rol de aula virtual, la cual permite gestionar los procesos de educación traspasando barreras de tiempo y distancia, es decir permiten acceder a sus tareas comunicarse con sus profesores y compañeros fuera del colegio, desde sus hogares o trabajos, porque tienen el material disponible 24/7, de esta manera al poder trabajar con el material fuera del colegio, los tiempos presenciales de los estudiantes en el aula es mejor utilizado ya que no se pierde tiempo en entrega o envío de tareas, siendo este uno de los principales objetivos que busca la institución.

Pretendiendo mejorar la interacción entre alumnos y maestros se propone el uso del gestor MOODLE como retroalimentación de las clases presenciales, el alcance del proyecto está dado en la implementación del LMS (Learning Management System – Sistema de Administración de aprendizaje), puesta en marcha y uso para la sección nocturna del Colegio Intisana. El cual se encuentra publicado en un host nacional de renombre, el Proyecto se realiza utilizando la metodología XP (Xtreme Programming).

Cuando se determinó la estructura y proceso con el que trabaja la institución se realizaron las configuraciones necesarias para el uso de alumnos y profesores de la entidad educativa, así como la creación de usuarios, y configuración de cursos a los que pertenecen cada uno de ellos, teniendo como objetivos específicos:

- Definir el estudio de factibilidad técnica.
- Instalar y configurar LMS acorde a procesos del Colegio Intisana.
- Capacitar a un súper usuario y al primer grupo que utilizará la herramienta.
- Implantar el sistema, para los cuartos y quintos cursos del Colegio.

Se ha planteado el problema ya detectado en el antecedente, mismo que se resolverá durante el desarrollo del presente proyecto, en función del diagnóstico, pronóstico y control del pronóstico.

¿Cómo implementar el gestor de aprendizaje Moodle como retroalimentación de las clases presenciales y mejorar la interacción alumno profesor, en cada asignatura?

Para la cual se espera una respuesta afirmativa al finalizar el proyecto de tesis, la misma que se sustentará con el cumplimiento de los objetivos planteados.

Se debe además abrir la mente a la realidad informática que se vive el día de hoy en el cual existe un mundo más allá de los apuntes y las aulas, un mundo que está transformándose a una velocidad increíble, el mundo virtual.

Internet es una de las mayores revoluciones que ha servido para difusión del conocimiento. Se cuestiona cada vez más el sistema tradicional de enseñanza y se plantean nuevas prácticas más activas y basadas en el logro de competencias que tengan en cuenta todo el proceso enseñanza-aprendizaje rompiendo barreras de tiempo y distancia.

Se precisa pues de una plataforma que permitirá integrar las diferentes posibilidades que nos otorga la red, de cara a su aprovechamiento en el ámbito educativo. Moodle es la más potente herramienta con la que cuentan los docentes en este momento para poder crear y gestionar cursos a través de la red. Permite subir contenidos educativos (apuntes, imágenes, videos, presentaciones, chat, entre otros), facilitar la comunicación entre maestro u alumno y entre alumnos y por último gestionar la evaluación de sus tareas de aprendizaje, es una nube que está en constante crecimiento, siendo cada vez más los profesores y alumnos de todo el mundo que lo conocen y lo utilizan.

MARCO TEÓRICO

2 MARCO TEÓRICO

Hace referencia a las teorías, e investigaciones realizadas en el área de interés, con el fin de sustentar el proceso de implementación de la solución tecnológica propuesta en el proyecto de tesis.

2.1 PLATAFORMA VIRTUAL

Son programas especializados que cuentan con herramientas, para la gestión de la enseñanza y el aprendizaje.

Intentan reproducir lo más fielmente posible la sensación de presencia.

Actores: Personas que de una u otra manera interactúan en la plataforma. (Miriam Vásquez Córdoba, 2012)

- Alumno
- Docente
- Gestión de alumnos
- Visitas
- Administrador de sistema
- Soporte técnico

2.1.1 LMS

Es un sistema de gestión de aprendizaje, instalado en un servidor, que se emplea para administrar, distribuir, controlar las actividades de formación no presencial de una institución como puede ser una plataforma virtual.

2.1.2 PROCESO ENSEÑANZA APRENDIZAJE

El proceso de Enseñanza Aprendizaje tiene dos actores principales; el profesor y el estudiante, acorde a Enrique Martínez-Salanova Sánchez el aprendizaje es “el cambio que se da, con cierta estabilidad, en una persona, con respecto a sus pautas de conducta. El que aprende algo, pasa de una situación a otra nueva, es decir, logra un cambio en su conducta”. (Enrique Martínez-Salanova, 2008)

Para realizar este cambio de conducta en el estudiante a través del aprendizaje se realizan varios procesos los cuales podemos identificar en la siguiente figura.



Figura 1. Proceso de Enseñanza Aprendizaje

En base a este proceso el objetivo de MOODLE es realizar los mismos pero de una manera mas dinámica y que ayude a realizarlos de mejor manera tanto para el profesor como para el estudiante.

2.1.3 EDUCACIÓN VIRTUAL

La educación virtual es una oportunidad y forma de aprendizaje que se acopla al tiempo y necesidad del estudiante. La educación virtual facilita el manejo de la información y de los contenidos del tema que se desea tratar y está mediada por las tecnologías de la información y la comunicación -las TIC- que proporcionan herramientas de aprendizaje más estimulantes y motivadoras que las tradicionales.

Este tipo de educación ha sido muy utilizada por estudiantes y profesores, además su importancia está incrementando, puesto que esta educación es una herramienta para incorporarnos al mundo tecnológico que será lo que próximamente predominará en la gran mayoría de los centros educativos. A través de ésta, además de la evaluación del maestro o tutor, también se puede evaluar constantemente nuestro propio conocimiento. (Fundación Wikimedia, Inc., 2012)

MOODLE como plataforma para la educación virtual permite que todos los usuarios tanto profesores como estudiantes, puedan hacer uso de sus ventajas para que de esta manera el procedimiento de aprendizaje sea mas ágil, intuitivo y hasta divertido.

2.2 APLICACIONES WEB.

Las interfaces amigables y la familiarización con Internet contribuyen a que el tiempo de aprendizaje se reduzca considerablemente respecto a aplicaciones tradicionales standalone.

El apogeo de soluciones o frameworks open source, hace que su desarrollo no sea limitado en cuanto a recursos de desarrollo y que un gran número de desarrolladores tengan experiencia con ellos.

2.2.1 MOODLE – Software educativo en Línea

MOODLE es un Ambiente Educativo Virtual, sistema de gestión de cursos, de distribución libre, que ayuda a los educadores a crear comunidades de aprendizaje en línea. Este tipo de plataformas tecnológicas también se conoce como LMS (Learning Management System).

Historia

MOODLE fue creado por Martin Dougiamas, quien fue administrador de WebCT en la Universidad Tecnológica de Curtin. Basó su diseño en las ideas del constructivismo en pedagogía que afirman que el conocimiento se construye en la mente del estudiante en lugar de ser transmitido sin cambios a partir de libros o enseñanzas y en el aprendizaje colaborativo. Un profesor que opera desde este punto de vista crea un ambiente centrado en el estudiante que le ayuda a construir ese conocimiento con base en sus habilidades y conocimientos propios en lugar de simplemente publicar y transmitir la información que se considera que los estudiantes deben conocer.

La primera versión de la herramienta apareció el 20 de agosto de 2002, a partir de allí han aparecido nuevas versiones de forma regular. Hasta julio de 2008, la base de usuarios registrados incluye más 21 millones, distribuidos en 46.000 sitios en todo el mundo y está traducido a alrededor de 91 idiomas. (Foster, H. y Cole, J., 2007).

Enfoque pedagógico

La filosofía planteada por MOODLE incluye una aproximación constructiva basada en el constructivismo social de la educación, enfatizando que los estudiantes (y no sólo los profesores) pueden contribuir a la experiencia educativa en muchas formas. Las características de MOODLE reflejan esto en varios aspectos, como hacer posible que los estudiantes puedan comentar en

entradas de bases de datos (o inclusive contribuir entradas ellos mismos), o trabajar colaborativamente en un wiki.

Habiendo dicho esto, MOODLE es lo suficientemente flexible para permitir una amplia gama de modos de enseñanza. Puede ser utilizado para generar contenido de manera básica o avanzada (por ejemplo páginas web) o evaluación, y no requiere un enfoque constructivista de enseñanza.

El constructivismo es a veces visto como en contraposición con las ideas de la educación enfocada en resultados, como en los Estados Unidos. La contabilidad hace hincapié en los resultados de las evaluaciones, no en las técnicas de enseñanza o en pedagogía, pero Moodle es también útil en un ambiente orientado al salón de clase debido a su flexibilidad.

2.2.2 HERRAMIENTAS Y PLATAFORMAS DE DESARROLLO.

El diseño de procesos y medición de datos se lo realizará con las siguientes herramientas tecnológicas.

2.2.2.1 Rational Rose

IBM Rational Rose Enterprise proporciona un lenguaje de modelado común que permite crear más deprisa software de calidad para el despliegue, diseño, construcción, pruebas y administración de proyectos en el proceso de desarrollo de software.

Es compatible con el lenguaje UML (Unified Modeling Language) y es uno de los productos más completos de la familia Rational Rose, permite el modelado UML para diseñar bases de datos, con la posibilidad de representar la integración de los requisitos de datos y aplicaciones mediante diseños lógicos y físicos. (Gertsch, Fredi, 2006)

2.2.2.2 Base de Datos

Las bases de datos MySQL y PostgreSQL son las bases de datos para desarrollo primario y de mejor utilización para MOODLE, las más completamente probadas y las que tienen abundante documentación y soporte. Oracle y MSSQL son completamente soportadas (se tiene que tomar en cuenta de que los plugins opcionales pueden estar no probados con estas bases de datos) pero la documentación y ayuda en línea no son tan completas como con MySQL/PostgreSQL.

MySQL tiene más documentación que cualquiera de las anteriores mencionadas, se necesita las extensiones PHP apropiadas (configuradas si lo requieren) para la base de datos elegida.

- MySQL - versión mínima 5.0.25
- PostgreSQL - versión mínima 8.3
- MSSQL - versión mínima 9.0
- Oracle - versión mínima 10.2
- SQLite - versión mínima 2.0

Se debe tomar en cuenta que Moodle no funciona bien con el ajuste "STRICT_TRANS_TABLES" de MySQL 5.x. Así que si esta es su base de datos deberá editar el fichero de configuración (*my.ini* en Windows o *my.cnf* en Linux/Unix) y comentar (o borrar) dicha opción. Necesitará reiniciar MySQL después de efectuar el cambio.

Normalmente el servidor de mysql escucha en el puerto 3306 o también se puede descargar un cliente de SQL por ejemplo mysql query browser o heidySQL se puede acceder al servidor de igual manera

2.3 METODOLOGÍAS

La fase de metodologías tanto de investigación con sus métodos y aspectos importantes; así como también se detallará la metodología informática orientada a objetos sus principales características y funcionalidades.

2.3.1 METODOLOGÍA DE INVESTIGACIÓN

Durante el desarrollo del proyecto de tesis, se utilizan métodos de investigación, los mismos que se describen a continuación:

2.3.1.1 MÉTODO DEDUCTIVO

Este método se lo ha empleado para recopilar información general sobre aplicaciones y herramientas de desarrollo web, metodologías de software e Informáticas; utilizando para ello libros y páginas web; obteniendo así un análisis y discernimiento de la información que es de utilidad.

2.3.2 METODOLOGÍA INFORMÁTICA

Las metodologías son necesarias para desarrollar cualquier proyecto de informática de manera ordenada y eficaz.

Programación Orientada a Objetos.

La idea principal de la programación orientada a objetos es hacer los procesos, herramientas y software reutilizable; es por eso que en base a las facilidades y ventajas que ofrece esta tecnología el sistema de este proyecto es un conjunto de objetos que interactuarán entre sí y que están organizados en clases donde se describen cada tipo de dato abstracto necesario en el programa y en consecuencia para la empresa.

Los sistemas orientados a objetos utilizan un modelo de ingeniería mediante proceso evolutivo también conocido como recursivo paralelo; que este tipo de

sistemas tienden a evolucionar con el tiempo, siendo esto muy factible para las empresas floricultoras ya que las formas de negocios constantemente cambian o se adecuan de acuerdo al mercado o clientes.

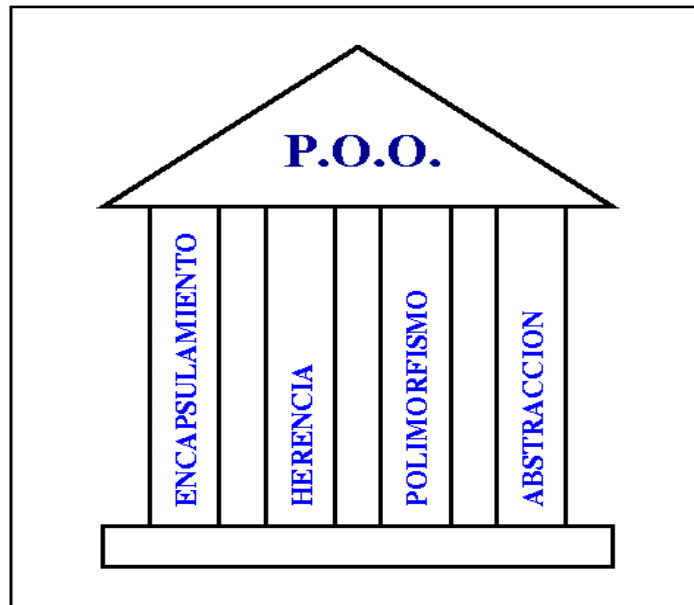


Figura 2. Pilares de la programación Orientada Objetos

2.3.3 METODOLOGÍA DE DESARROLLO

Es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información. Programación Extrema (Extreme Programming - Xp).

Programación Extrema o Extreme Programming es una metodología ágil de desarrollo de software. Se diferencia de las otras metodologías debido a que pone más énfasis en la adaptabilidad que en la previsibilidad, se centra en las personas más que en los procesos.

La programación extrema busca dos objetivos: hacer un software de calidad y de la manera más rápida posible. Forma parte del movimiento de desarrollo ágil de software, que se basa en la adaptabilidad de cualquier cambio como medio para aumentar las posibilidades de éxito de un proyecto.

Esta metodología trata de dar al cliente el software que necesita y cuando lo necesita. Por tanto, se debe responder inmediatamente a las necesidades del cliente, incluso cuando los cambios sean al final del ciclo de la programación. También trata de potenciar al máximo el trabajo en grupo. Tanto los jefes de proyecto, clientes y desarrolladores, son parte del equipo y están involucrados en el desarrollo del software.

Principios

XP como metodología de desarrollo ágil tiene como principios:

- Las personas y sus interacciones son más importantes que los procesos y las herramientas.
- Un sistema funcional es más importante que la documentación exhaustiva.
- La colaboración del cliente en lugar de la negociación de contratos.
- La respuesta delante del cambio en lugar de seguir un plan cerrado.

La programación extrema se basa en la simplicidad, comunicación y reciclado continuo de código. Presenta muchos puntos comunes con el desarrollo incremental, comenzando con el hecho de que el software desarrollado con XP se realiza de forma incremental. La disciplina de XP se puede dividir en cuatro fases, las cuales se definen a continuación:

Codificar. Definimos a codificar como el proceso de plasmar las ideas de la interpretación del problema en código fuente y que proporcione beneficios al cliente. Por lo tanto, el software se produce a través de pequeñas versiones incrementales de producción corta.

Probar. Hacer pruebas permite saber que el trabajo funciona correctamente. Lo mejor es generar las pruebas antes de generar el código fuente, es decir, desde el momento en que se conocen los casos de uso o, según la metodología XP, las historias del usuario, estas serán revisadas más adelante.

Escuchar. Si se van a hacer pruebas se debe preguntar si lo obtenido es lo deseado, encontrar cuales son los objetivos a cumplir, y se lo debe hacer a la persona que va a hacer uso de la herramienta. Se debe escuchar a los clientes cuales son los problemas del negocio, explicando lo que es fácil o difícil de obtener, la retroalimentación entre ambos ayuda en gran manera a entender los problemas.

Diseñar. El diseño consiste en crear una estructura lógica del sistema. Este también debe ser incremental y su estructura debe ser clara, si hay alguna parte compleja, se la debe dividir en varias partes.

Filosofía y Valores

A la programación extrema se la considera una metodología un tanto filosófica, por los valores que promueve en el grupo de trabajo y durante la realización de proyectos.

Los valores en los que la metodología hace un planteamiento más global que el simple hecho del desarrollo de software son:

Comunicación. Los programadores se comunican a través del código fuente, mientras más simple sea este, mejor es su comunicación. Se trabaja en grupos de dos personas por computador pero con total comunicación en todo momento con los otros grupos. El código autodocumentado es más fiable que los comentarios porque estos quedan desfasados a medida que el código es modificado. Se debe documentar únicamente lo que no va a variar en el tiempo, como el objetivo de una clase o la funcionalidad de un método. Otra manera de comunicación son las pruebas unitarias, ya que describen el diseño de las clases y métodos al mostrarse como deben ser utilizadas en ejemplos específicos. La comunicación con el cliente es importante, esta debe ser fluida por parte del equipo de desarrollo, ya que el cliente decide qué características son importantes y ayuda a solucionar dudas que los desarrolladores puedan tener. (Schenone Marcelo Hernán, 2004).

Simplicidad. Lo más simple es lo mejor, funciona mejor, es más rápido, más adaptable, más barato y además es más entendible. Se simplifica el diseño para que sea más fácil de desarrollar y de esa manera facilitar el mantenimiento. Para mantener la simplicidad es necesario la refactorización de código, de manera que el código se mantenga simple a medida que crece. La simplicidad también se aplica a la documentación, tratando de que la mayor parte del código sea autodocumentado y colocando adecuadamente los nombres de variables, métodos y clases, de manera que estos sean intuitivos para todo el grupo de trabajo. La simplicidad y la comunicación se complementan, cuanto más simple el sistema menos se debe comunicar de él. (Schenone Marcelo Hernán, 2004).

Retroalimentación (feedback). Por medio de pruebas funcionales al sistema, este mantendrá informado al equipo del grado de fiabilidad que posee. El continuo contacto con el usuario, al irle entregando las sucesivas versiones en funcionamiento del producto, permite que este de su valoración y comunique,

cada vez mejor, lo que realmente quiere del sistema. La retroalimentación actúa junto con la simplicidad y la comunicación, cuanto mayor retroalimentación más fácil es la comunicación. Cuanto más simple un sistema más fácil de probar.

Valentía. Para los gerentes la programación en parejas puede ser difícil de aceptar porque tienen la idea de que la productividad se va a ver afectada puesto que solo la mitad de los programadores está escribiendo código. Hay que ser valiente para confiar en que la programación por parejas favorece la calidad del código sin que la productividad se vea afectada negativamente. Se requiere valentía para implementar las características que el cliente quiere ahora sin caer en la tentación de optar por un enfoque más flexible que permita futuras modificaciones. No se debe emprender el desarrollo de grandes marcos de trabajo (frameworks) mientras el cliente espera. En ese tiempo el cliente no recibe noticias sobre los avances del proyecto y el equipo de desarrollo no recibe retroalimentación para saber si va en la dirección correcta. (Schenone Marcelo Hernán, 2004).

Respeto. Una manera de demostrar respeto en el grupo de trabajo es que los programadores no realizan cambios que puedan hacer que las pruebas existentes fallen o que demoren el trabajo del resto de sus compañeros. Todos los miembros del grupo siempre están buscando que el diseño del sistema sea óptimo de tal manera que se obtenga alta calidad del producto.

YAGNI. “You aren’t gonna need it” (No lo vas a necesitar). No se debe hacer nada que se crea que en el futuro pueda ser útil porque probablemente no lo van a necesitar, es simplemente una pérdida de tiempo. (Schenone Marcelo Hernán, 2004).

OAOO. “Once and only once” (Una única vez). Las cosas se hacen una sola vez, probablemente el punto que aporte un mayor valor como aspecto filosófico es el de la comunicación, puesto que logra que los miembros del proyecto se comprometan más con el cumplimiento del mismo.

Además, la filosofía del “pair programming” es valiosa en otros aspectos de la vida cotidiana, ya que ayuda a que la gente acostumbrada a trabajar así, sea más receptiva a las diversas situaciones que se les presenta. También cobra especial importancia el hecho de que el cliente se considera como un elemento más del equipo, ya que es el encargado de ofrecer la retroalimentación diariamente y de crear las denominadas historias del usuario, algo parecido a los casos de uso.

Ventajas

La utilización de la metodología de la programación extrema proporciona varias ventajas a la hora de afrontar varios proyectos, puesto que en ciertos contextos puede ahorrar mucho tiempo y recursos.

Ciertamente, para que la metodología de la programación extrema esté siendo tomada tan en cuenta, es porque debe ofrecer una serie de ventajas al momento de utilizarla, que hagan que el esfuerzo de entender y aplicar sus prácticas, sea insignificante en comparación a los beneficios obtenidos.

Se obtienen productos utilizables rápidamente, se tiene un proceso de integración continuo, por lo que desaparece el esfuerzo final de integración, consiguiendo integrar todo el trabajo de manera mucho más fácil.

Las necesidades del usuario son atendidas con mayor precisión. Esto es posible gracias a las continuas versiones que se entregan para el usuario.

Los productos generados son más fiables y robustos contra los fallos, debido al diseño de las pruebas realizado antes de la codificación, el código es más simple y de fácil comprensión, reduciendo de esa manera el número de errores.

Gracias a la filosofía de la “Programación en parejas”, es posible que los desarrolladores apliquen las buenas prácticas que ofrece XP.

Con la refactorización de código, es más fácil modificar los requerimientos del usuario.

Cualquier persona del equipo puede desarrollar, mejorar, simplificar, cualquier requerimiento del proyecto, utilizando siempre sistemas CSV para evitar la duplicidad en el trabajo y usando la refactorización de código si se trata de una modificación. Esto es posible debido a que la propiedad del código es colectiva.

La metodología XP puede brindar más ventajas de las antes mencionadas, según el entorno en la cual se aplique, la programación extrema no debe ser considerada como si fuese una herramienta que permitiese solventar cualquier tipo de situación. A continuación se intenta mostrar en qué situaciones y bajo qué criterios es mejor aplicar la programación extrema.

2.3.3.1 Visión general de la programación extrema

Los pilares de la programación extrema, son un conjunto de 12 buenas prácticas, a las cuales se las puede denominar como los 12 mandamientos de la programación extrema, estos son:

La planificación: utiliza la “historias del usuario” para realizar el análisis, dividiéndolas en tareas (unidades pequeñas, de 1 a 5 días de trabajo en pareja), se priorizan las tareas y cada una de ellas será desarrollada de manera incremental.

Versiones pequeñas: La primera versión tendrá los requisitos mínimos, más útiles y necesarios para el sistema global.

Sistema metafórico: Cada proyecto debe poseer una metáfora asociada que ofrezca criterios para nombrar lo que se vaya haciendo de manera fácil.

Diseño simple: Debido a que los requerimientos cambian o pueden cambiar con frecuencia, hay que utilizar los diseños más simples para cumplir con los requerimientos que se tienen en ese momento.

Pruebas continuas: Antes de implementarse cualquier característica de un sistema, se debe escribir una prueba para ella.

Refactorización: Si se debe agregar una nueva característica al sistema, y esta tiene mucho en común con otra ya existente, es recomendable eliminar el código duplicado, sin temor a fallos, porque las pruebas validarán el correcto funcionamiento.

Programación en parejas: Se trabaja en parejas, cada una utilizando un único equipo, de esa manera el código es revisado mientras se desarrolla.

Propiedad colectiva del código: Cualquier persona del equipo de trabajo puede modificar cualquier módulo en cualquier momento, nadie es dueño de nada.

Integración continua: Todos los cambios se introducen al sistema, al menos una vez al día.

Semanas de 40 horas de trabajo: No se deben trabajar horas extras.

Cliente en su sitio: Se debe tener un usuario del sistema que siempre se encuentre disponible para los miembros del equipo de trabajo.

Estándares de codificación: Todos los miembros del equipo de desarrollo deben utilizar los mismos criterios para programar, así no será posible determinar quién realizó una parte específica de la implementación.

2.3.3.2 Análisis y diseño en programación extrema

Las tareas típicas previas al desarrollo de código como son el análisis y el diseño son tareas muy importantes en la programación extrema, pero se realizan con un enfoque más ligero y transparente. El análisis es parte fundamental, con él se intentan recoger todas las necesidades del usuario. De él surgen las “historias del usuario” que servirán para empezar a desarrollar el sistema.

2.3.3.3 Historias del usuario

Este concepto está ligado a los famosos casos de uso, utilizados en el desarrollo incremental del software. Su similitud se basa en que el cometido y funcionalidad son los mismos, pero no son iguales. Permiten sustituir requerimientos extensos por una serie de historias del usuario y además permite estimar el tiempo para la reunión de lanzamientos de futuras versiones del sistema.

Además de esto, las historias del usuario ayudan a crear pruebas que se aplicarán al sistema para ver si cumplen cada una de ellas, lo cual es equivalente a cumplir un determinado requisito en otros modelos de desarrollo.

Son realizadas por el propio cliente en forma de tres sentencias de texto, en las que describe necesidades del sistema con la propia terminología del negocio, sin hacer uso de vocabulario técnico. (Begoña Alfareme González, 2002).

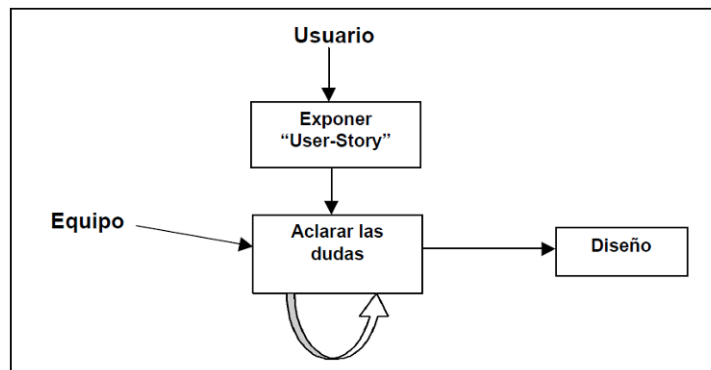


Figura 3. Historias del usuario

2.3.3.4 Modelos de negocio

El proyecto de software debe estar perfectamente alineado con el modelo de negocio para el que fue creado. El problema surge debido a que los modelos de negocio cambian cada vez más rápido, ya que las empresas deben actuar cada vez contra más elementos y más variados (respuestas ante nuevas tecnologías, adecuar los procesos de negocio a estas, razones de competitividad, entre otros). Debido a esto, el sistema de desarrollo de software debe ser capaz de responder de una manera lo suficientemente rápida a los cambios en las necesidades del cliente.

2.3.3.5 Interacción entre las cuatro variables de la gestión de proyecto

Tabla 1. Interacción entre las cuatro variables de la gestión de proyecto.

Variable	Aumento en exceso	Disminución
Alcance		Mejora la calidad, siempre y cuando se resuelva el problema básico del cliente. Permite reducir coste y tiempos de entrega.
Tiempo	Puede mejorar la calidad y el alcance, pero en exceso puede ser perjudicial. La mejor retroalimentación viene del sistema en producción.	Si es corto, la calidad del sistema se verá afectada e inmediatamente el alcance, el tiempo y el coste.
Coste	Más dinero puede engrosar el sistema pero en exceso puede crear más problemas que los que resuelve.	Con poco dinero, será imposible resolver todas las necesidades del cliente.
Calidad	Exigir mayor calidad permite conseguir plazos más cortos o hacer más en un tiempo determinado. Las personas trabajan mejor si sienten que hacen un buen trabajo.	Se puede sacrificar para obtener mayores ganancias a corto plazo, pero los costes posteriores son enormes (humanos, de negocio y técnicos).

2.3.3.6 Actores y funciones en programación extrema

En un proyecto de software existen diferentes roles (actores) y responsabilidades para cada tarea, la programación extrema también requiere de estos elementos durante las distintas etapas del proceso.

Jefe del proyecto

El jefe del proyecto es la cabeza del proyecto, posee una serie de responsabilidades importantes, siendo una de las más importantes el hacer que se cumpla con todo lo previsto, siendo además el defensor de los derechos del cliente, es quién controla que se cumplan ciertos requisitos.

Brinda al cliente una visión global del sistema, indicando lo que se va hacer, cuándo se inicia y el valor que este va a tener.

Procura, por cada semana de trabajo, entregar la mayor parte de tareas implementadas que sean posibles, además ofrece dinamismo al cliente, para que este vea que el sistema evoluciona.

Además genera con el cliente un ambiente de respeto y colaboración lo cual se convierte en un ambiente dinámico de trabajo, en el cual el sistema o implementación evoluciona de una manera correcta y sin trabas.

Mantiene informado al cliente del cronograma del proyecto, así como de los posibles retrasos, para que este pueda realizar las modificaciones que considere pertinentes si su afán es conservar el cronograma original.

Algunas de las funciones que posee el jefe de proyecto son: Causar, Coordinar, Reportar, Recompensar, Eliminar obstáculos, entre otras. A continuación se define estas funciones para tener una visión más clara de cada una de ellas.

Causar. Significa conseguir que las cosas ocurran, siendo el desencadenante de todos los procesos necesarios para el desarrollo del proyecto. Por esa razón debe estar alejado del desarrollo del proyecto para mantener una posición objetiva. Debe estar más de parte del cliente que del desarrollador, corrigiendo todo lo que considere sea un fallo, o que interrumpa el flujo normal del proceso.

Coordinar. Es quien debe distribuir, en cierta medida, la carga de trabajo a cada miembro del equipo, controlar que el equipo se encuentre bien, actuando de intermediario, para arreglar cualquier problema que pueda surgir entre los miembros del equipo.

Reportar. En la programación extrema, la documentación más valiosa es aquella en la que se definen cuantas historias del usuario han sido implementadas en la versión actual y cuáles serán implementadas en la siguiente.

Recompensar. Debe planificar las recompensas que se ofrecerán en caso de que el proyecto sea un éxito absoluto. A cualquier trabajador le gusta verse recompensado cuando su trabajo es hecho de la mejor manera.

Eliminar obstáculos. Intenta que los desarrolladores no tengan que preocuparse por cualquier cosa ajena a su trabajo como tozudez del usuario, y se dediquen únicamente a su labor.

El cliente

El cliente debe estar en todas las etapas del proyecto, es el objetivo y a quien se debe satisfacer. Durante el análisis, es quien realiza las historias del usuario, a quien se le debe presentar las pruebas para que éste de su aprobación, quien recibirá todas las versiones del sistema que se vayan generando.

Además, el cliente es quien va a solventar todas las dudas que se generen durante el desarrollo, es el único que posee el conocimiento requerido para solventar esas dudas.

Los programadores

Sin un equipo de desarrollo no se podría realizar ningún proyecto de software. En la programación extrema, los programadores son los que realizan el análisis, diseñan, programan y realizan las pruebas. Por todo eso, la clave para el éxito de un proyecto es la coordinación entre el cliente y los programadores.

2.3.3.7 Fases de la programación extrema

El ciclo de vida de la programación extrema se enfatiza en el carácter iterativo e incremental del desarrollo. Una iteración del desarrollo es un periodo de tiempo en el que se realizan un conjunto de funcionalidades o historias del usuario.

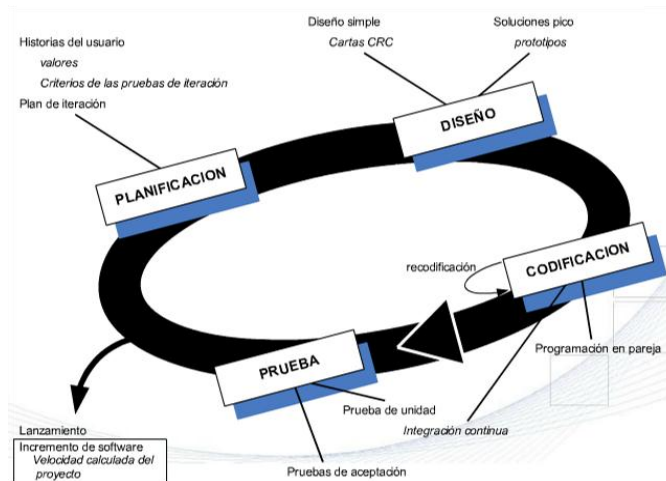


Figura 4. Fases de la metodología de XP

Las iteraciones son cortas puesto que la programación extrema parte del hecho de que, mientras se generen versiones, más retroalimentación se va a obtener por parte del usuario y eso implica una mejor calidad del producto.

Cada iteración incluyen las fases de diseño, codificación y pruebas. Estas fases son superpuestas de tal manera que no se separan en el tiempo.

Planificación

El cliente tiene un papel importante de interacción con el equipo de programadores, en cada una de sus etapas, después de cada cambio y de cada posible problema, identificando prioridades y expresando su criterio.

Debido a que el cliente se encuentra muy cercano al proceso de desarrollo, se elimina la fase de recopilación de requerimientos y se permite que estos se vayan recolectando a lo largo de todo el proyecto, de manera ordenada. Esto posibilita al cliente a hacer cambios sobre la marcha, pero eso trae consigo que siempre se encuentre disponible para solventar las dudas que el equipo de desarrollo pueda presentar.

El cliente debe crear las historias del usuario. Estas constan de dos fases:

- En la fase uno, el cliente describe en sus palabras las características que debería poseer el sistema. El jefe del proyecto es el encargado de informarlo de las dificultades técnicas de cada una de ellas y su coste. Como resultado de este diálogo, el cliente deja por escrito un conjunto de historias del usuario, ordenadas de manera prioritaria para él. De esta manera es posible definir fechas aproximadas de entrega.
- En la fase dos, el cliente toma cada una de las historias del usuario a implementar y las dividirá en trabajos a realizar, esto da como resultado una planificación más exacta.

La programación extrema exige que el cliente sea quien redacte los documentos con las especificaciones de lo que realmente quiere, como un documento de requerimientos de usuario.

Durante este proceso, el equipo técnico será el encargado de catalogar las historias del usuario y asignarles una duración. La regla es que cada historia del usuario debe tener una duración máxima entre una y tres semanas de desarrollo. Las historias más cortas serán agrupadas mientras que las otras serán modificadas o divididas.

Las historias del usuario serán escritas en tarjetas, lo que facilitará que el cliente las ordene por prioridad y que los programadores las cataloguen correctamente. También sirven al momento de realizar las pruebas de aceptación.

La programación extrema plantea la planificación como un permanente diálogo entre el cliente y el equipo técnico. Cada uno de los actores tiene sus funciones y responsabilidades dentro de la planificación.

El cliente debe determinar:

- **Ámbito.** ¿Qué es lo que realmente el software debe resolver para que este genere valor?
- **Prioridad.** ¿Qué debe realizarse en primer lugar?
- **Composición de versiones.** ¿Qué debe incluir en cada una de ellas? ¿Cuánto es necesario hacer para saber si el negocio va mejor con él que sin él? Una vez que el sistema aporte algo, se debe generar las primeras versiones.
- **Fechas de versiones.** ¿Cuáles son las fechas en las que el sistema o parte del mismo generen valor al negocio?

El equipo técnico es responsable de:

- Estimar el tiempo requerido para implementar las funcionalidades detalladas por el cliente.
- Informar sobre las consecuencias de determinadas decisiones por parte del cliente.
- Organizar la cultura de trabajo.
- Realizar la planificación detallada de cada versión.
- Verificar que los tiempos de desarrollo o implementación se cumplan.
- Dar un mantenimiento preventivo a las maquinas que utilizaran la implementación para garantizar su buen funcionamiento.

Adicional a esto, el equipo técnico debe tomar en consideración algunos aspectos que permitirán que el proyecto fluya de manera sana y sin contratiempos, estos son:

- El equipo técnico no empezará a realizar el análisis hasta que no tengan las historias del usuario.
- Los desarrolladores deben regirse únicamente en lo que se encuentre en el diseño y nunca deberán tomar decisiones que no consten en él.
- El equipo de trabajo nunca deberá incluir funcionalidades que no se encuentren en el diseño. Esto es porque estas posiblemente no sean de total agrado para el cliente y se requiera hacer cambios en un código que nunca debió haber existido, esto es un trabajo inútil y no reconocido por lo que es mejor no salirse de especificaciones.

La manera de planificar bien el trabajo para la programación extrema, es de la siguiente forma:

- Redactar las historias del usuario
- Crear el plan de entregas
- Controlar la velocidad del proyecto
- Dividir un proyecto en iteraciones
- Planificar cada iteración antes de comenzarla
- Rotar al personal
- Realizar reuniones de seguimiento del proyecto diarias
- Corregir la propia metodología

Diseño

El diseño adecuado para el software es aquel que funciona con todas las pruebas, no tiene lógica duplicada, manifiesta cada intención importante para los programadores y tiene el menor número de clases y métodos. Algunos puntos que ayudan a obtener el diseño esperado son:

Metáfora. Es una historia que todos pueden contar acerca de cómo funciona el sistema. Esta ayuda a entender a cualquier persona el objeto del programa.

Diseño sencillo. La metodología de programación extrema sugiere que hay que hacer todo lo menos complicado posible para conseguir un diseño fácil de implementar, cuyo desarrollo posea un coste menor en tiempo y esfuerzo.

Glosario de términos. El uso de un glosario de términos y una correcta especificación de los nombres de métodos, clases y variables, ayuda a comprender mejor el diseño y facilita su escalabilidad y reutilización de código.

Tarjetas C.R.C. (Clase, Responsabilidad y Colaboración). Permiten al programador tener un desarrollo orientado a objetos.

Desarrollo

En la programación extrema, la presencia del cliente en todas las fases es indispensable. Al momento de codificar una historia del usuario es aún más necesaria. Se debe recordar que los clientes son los que crean las historias del usuario y negocian los tiempos en que serán implementadas.

Esta etapa debe reunir las siguientes características:

Refactorización. Cuando se implementan nuevas características a los sistemas se busca la manera de hacerla lo más simple posible, después de implementada, surge la incógnita de cómo hacerla más simple sin que pierda funcionalidad. Este proceso es la refactorización, esto se realiza únicamente cuando el sistema lo pida.

Programación por parejas. Cada miembro tiene un papel: uno codifica y piensa la mejor manera de hacerlo, y el otro piensa más estratégicamente. Cualquier miembro del equipo se puede emparejar con cualquiera, si se tiene un trabajo sobre un área que no se conozca a fondo, se empareja con otra persona que posea el conocimiento necesario.

Propiedad colectiva. Cualquier miembro del equipo que sepa que puede aportar valor al código en cualquier parte del desarrollo puede hacerlo. Ningún miembro del equipo es propietario del código.

Integración continua. El código debe ser integrado por lo menos una vez al día y se debe realizar las pruebas sobre la totalidad del sistema.

40 horas de trabajo. Se trata de que el equipo de desarrollo se encuentre fresco y motivado cada mañana y cansado y satisfecho cada noche. No se deben trabajar horas extras.

Cliente in situ. El cliente debe sentarse con el equipo de programadores, estar disponible para responder a sus preguntas, resolver discusiones y fijar prioridades.

Estándares de codificación. Es importante establecer un estándar de codificación aceptado e implementado por todo el equipo con el fin de que cualquiera de los miembros del equipo esté en capacidad de entender el código y poder realizar cualquier modificación al mismo.

Pruebas

Uno de los pilares de la programación extrema son las pruebas o test. Las pruebas son las encargadas de verificar el funcionamiento del programa que se está implementado.

Se deben crear aplicaciones que serán las encargadas de realizar las pruebas con un entorno de desarrollo específico para cada prueba. Se deben crear las pruebas que se van a realizar antes de implementar el código.

En el repositorio debe constar el código acompañado de las pruebas que lo verifican, de esta forma se asegura el uso colectivo del código.

Las pruebas permiten verificar que un cambio en la estructura de un código no cambia en nada la funcionalidad del mismo.

Las pruebas o test de aceptación sirven para evaluar las distintas tareas en las que ha sido dividida cada una de las historias del usuario. Estos son creados y

usados por los clientes para validar que las distintas historias del usuario cumplan con su objetivo.

Las pruebas serán realizadas sobre las funcionalidades generales que debe cumplir el programa, según lo especificado en la descripción de requisitos. Esto da como resultado un programa más seguro que soporte cambios en el tiempo.

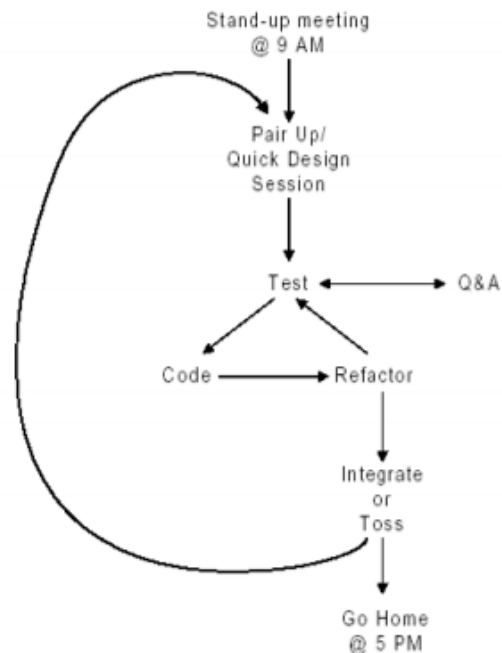


Figura 5. Un día de trabajo con XP

2.3.3.8 Comparativa con otras metodologías tradicionales

La programación extrema es criticada en gran manera especialmente por su principio de la programación por parejas. Muchos jefes de proyectos no están de acuerdo con este punto porque piensan que es un desperdicio del recurso humano y los desarrolladores posesivos creen que ellos son los mejores conocedores de las herramientas y lenguajes de programación que utilizan. Se cree que esto solo puede funcionar con programadores expertos, que son capaces de hacer un buen diseño, simple y flexible.

Puesto que la programación extrema es totalmente opuesta a las metodologías tradicionales, en las que se impone que los proyectos de software estén debidamente documentados, volviendo el trabajo demasiado burocrático e ineficaz, XP se vuelve más liviana y está orientada a las personas más que a los procesos.

Esto ha provocado mucha controversia dentro de la ingeniería de software. A continuación, se muestra una gráfica de un *análisis realizado por IBM*, esta grafica fue sacada de un estudio estadístico realizado en el año 2010, debemos tomar en cuenta que cada año el uso de las metodologías rápidas va en ascenso.

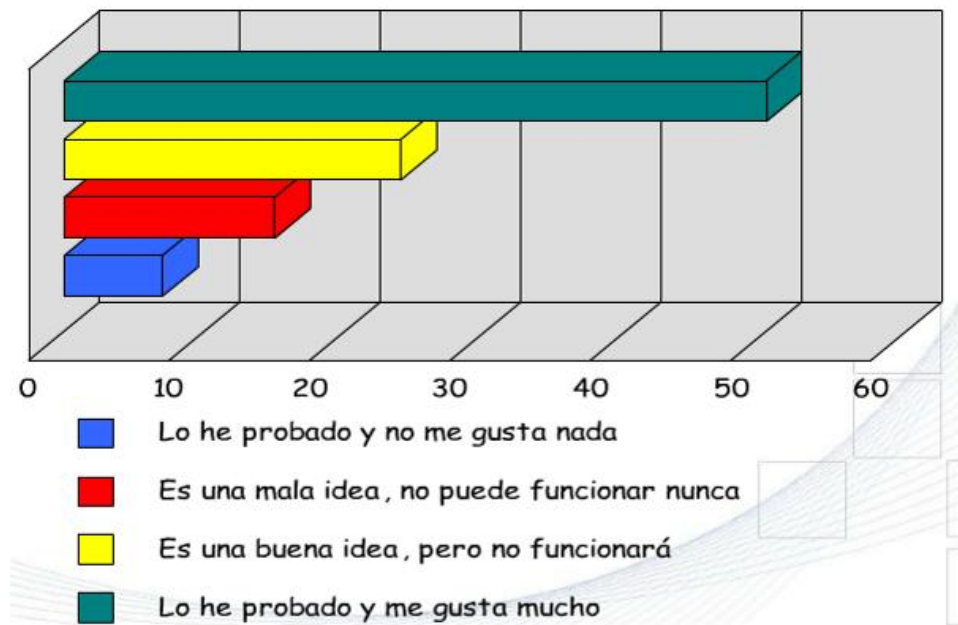


Figura 6 Comparativo de XP con otras metodologías

La programación extrema está convencida que:

- Las personas son el punto clave en todos los procesos de desarrollo.
- Los programadores son profesionales y no necesitan ser supervisados.
- Los procesos se aceptan y se acuerdan, no se imponen.
- El liderazgo del proyecto es compartido por los desarrolladores y gerentes.
- Se tiene un trabajo conjunto entre desarrolladores y las personas que conocen del negocio.
- El cliente es parte principal en la implementación de cualquier solución.
- La generación de código o implementación de cualquier solución es más importante que la documentación.
- Es de suma importancia generar versiones de ser posible semanales para que el cliente pueda verificar que se está yendo en el camino correcto a sus necesidades.

2.3.3.9 Beneficios y desventajas de la programación extrema

La programación extrema apunta a ahorrar tiempo y dinero mientras ayuda a los programadores a crear aplicaciones de mejor calidad. Los programadores son inducidos a trabajar en ciclos de desarrollo más rápidos y múltiples reiteraciones, lo que da flexibilidad para ajustar las aplicaciones a través de todo el ciclo de vida del desarrollo.

La integración del cliente dentro del avance de la aplicación ayuda de gran manera a que tanto los programadores no pierdan tiempo en desarrollos que pueden ser descartados y que el cliente este de acuerdo con que el sistema sustentara de ayuda a los requerimientos que tiene.

Los Beneficios y desventajas que se puede tener al utilizar la metodología de Programación Extrema son los que se listan a continuación:

Beneficios

- Satisfacción del cliente.
- Mejora la comunicación.
- Cumplimiento de tiempos.
- El cliente es el que establece prioridades.
- Introduce eficiencia en la planificación y pruebas.
- Ofrece un trabajo de calidad.
- Esta metodología es muy buena aplicada en la implementación de nuevas tecnologías donde los requerimientos cambian muy rápidamente.

Desventajas

- No siempre puede ser más fácil que el desarrollo tradicional.
- Los clientes puede ser que no quieran tener constantemente pequeños desarrollos sino que esperen ver ya el trabajo terminado.
- Esta metodología requiere que todos se ajustes de manera rígida a sus principios.

METODOLOGÍA

3 METODOLOGÍA

Las técnicas para el desarrollo de software así como para su implementación son un conjunto de pasos los cuales ayudan a llevar un orden lógico el cual permite culminar todos los objetivos generando un producto de calidad.

Existen varias metodologías las cuales ayudan a cumplir con la meta del desarrollo y la implementación de un sistema, puesto a que el colegio Intisana necesitaba de urgencia el sistema en producción antes del inicio de clases, se decidió basarse en una metodología rápida de desarrollo para escoger la metodología que mejor se adapte a la implantación se hizo un análisis a la tesis “Análisis comparativo de los metodologías existentes orientadas al desarrollo ágil de aplicaciones”, (Verónica Rodríguez, 2011), realizada en la Universidad Tecnológica Equinoccial, con este análisis y una encuesta para confirmar la metodología a utilizar se escogió XP (extreme Programming), la misma que tiene como característica principal de sus etapas, ser:

- claras
- optimas

Esto ayuda para realizar un proceso rápido en base a la cooperación entre usuario y programador.

Las facetas de esta metodología son:

- Planificación
- Diseño
- Codificación
- Pruebas

Las mismas que de detallan a continuación.

Planificación.- en esta etapa se plantea la factibilidad de la implementación del sistema en la institución, en resumen determinamos si la implantación del sistema en realidad ayuda a la mejora de los procesos y si esto es factible en un análisis costo beneficio, de ser positivos los resultados de estos análisis, se continua con la siguiente paso.

Se toma en cuenta que la factibilidad e implantación del sistema se lo verifica técnicamente es decir: que software y hardware es el recomendado para su buen funcionamiento en comparación al que se va a utilizar.

En la factibilidad económica se tiene un cuadro de cuanto es el coste de la implantación del sistema tomando en cuenta software hardware y horas de desarrollo. La factibilidad operativa dará un resumen de uso del sistema según los tipos de usuario que lo vayan a utilizar. Para finalizar este análisis la factibilidad legal permitirá ver como se aplicaría el sistema en los procesos de la institución acorde a sus reglamento interno.

Diseño.- En la parte de Diseño se refiere a dos temas El diseño de Interfaz de usuario, y el diseño de la Base de Datos, en el primero se maqueta las ideas principales de como se definen las pantallas para el uso de los usuarios de la aplicación, esta maquetación no tiene color ni logos es decir se puede identificar la accesibilidad en funcionamiento de las pantallas, en la parte del diseño de la Base de Datos se puede observar el modelo entidad relación lo cual dará idea de como esta estructurada la información en el sistema moodle.

Codificación.- La fase de codificación en la cual se muestra paso a paso como se realiza la creación de los principales elementos para el funcionamiento del sistema en su inicio estos son los pasos que se siguieron para la implantación del sistema en su servidor dedica, los pasos para la creación de cursos y finalmente los pasos para la creación de usuarios, tomando en cuenta que los

módulos principales de MOODLE ya vienen instalados en la parte de implantación del mismo.

Pruebas.- en la parte de pruebas se utiliza a los tres tipos de usuarios para corroborar la viabilidad de la factibilidad operativa en el entorno de moodle, una vez realizadas estas pruebas y verificando que no existan errores tanto de diseño como de itinerancia a la página, para esto se utiliza el sistema Webserver Stress Tool el cual permite realizar pruebas las cuales muestran como se comportara el dominio simulando con diferentes patrones el ingreso de varios usuarios al mismo tiempo así como el uso de información de internet, tanto de subida como de bajada con carga de información, cookies, claves, es decir sobrecargando la pagina para comprobar su resistencia a la carga de información.

ANÁLISIS Y RESULTADOS

4 ANALISIS Y DISCUSIÓN DE RESULTADOS

El propósito de la presente investigación fue detectar los procesos entre profesor y estudiante en los cuales el sistema Moodle pudiera dar fuerza, agilizar y ayudar a los estudiantes de la sección nocturna del Colegio Intisana de Quito.

Para implementar estas soluciones se aplicó la configuración del servidor en donde se realizó la instalación de MOODLE y los módulos que sean necesarios para alcanzar los objetivos planteados.

4.1 FASE INICIAL – ANÁLISIS

4.1.1 INTRODUCCIÓN

4.1.1.1 Propósito

El presente documento está dirigido a todas la personas con conocimiento informático que deseen una guía del desarrollo de la investigación de procesos de una entidad educativa, a la cual deseen implementar el sistema moodle, como ayuda informática para la agilización de los mismos. Esta guía permitió a través de la utilización de la metodología XP identificar los pasos para una implementación del sistema moodle.

4.1.1.2 Ámbito del Sistema

El sistema MOODLE brindó opciones las cuales ayudaron a interactuar entre profesor y alumno, estas opciones son: chat, blog, carga de pruebas y notas, foros, entre otros. Tomando en cuenta que en MOODLE no se puede dar clases en tiempo real, para estos casos se utilizó una aplicación diferente como complemento para MOODLE.

La aplicación de MOODLE en la institución ayudó a agilizar la entrega de trabajos, deberes, pruebas y de esta manera los estudiantes y profesores de la sección nocturna de la institución pudieron aprovechar de mejor manera el tiempo presencial.

A futuro la expectativa es que el sistema se encuentre trabajando en toda la institución es decir incluir la sección diurna, y crear nuevos tipos de usuarios como podría ser el padre del estudiante.

4.1.1.3 Definiciones, acrónimos y abreviaturas

- ERS: Especificaciones de requisitos de Software.
- CU: Casos de USO.
- XP: Metodología rápida de desarrollo -Xtreme Programming-.
- FCI: Facultad de Ciencias de la Ingeniería.
- UTE: Universidad Tecnológica Equinoccial.
- ITT: Investigación y transferencia de tecnología.

4.1.1.4 Referencias

IEEE Recommended practice for software Requirements Specification IEEE STD 830-1998.

4.1.2 DESCRIPCIÓN GENERAL

4.1.2.1 Perspectiva del producto

Se implementó el producto MOODLE el cual es un LMS() diseñado para la interacción entre profesores y estudiantes principalmente, creando un ambiente

de aula virtual en la cual los usuarios pueden realizar actividades comunes del proceso de estudio sin necesidad de una presencia física.

4.1.2.2 Funciones del producto

Las principales funciones que el sistema posee son las listadas a continuación.

- Configuración de curso.
- Subir Novedades.
- Organizar archivos.
- Agregar recursos.
- Agregar tareas (Fotos, tareas, cuestionarios).
- Revisar tareas, foros y participaciones.
- Estimular la atención y motivar.
- Estimular el recuerdo de los conocimientos y habilidades previas, esenciales y relevantes.
- Presentar el material a aprender.
- Guiar y estructurar el trabajo del aprendiz.
- Provocar la respuesta.
- Proporcionar feedback.
- Promover la generalización del aprendizaje.
- Facilitar el recuerdo.
- Evaluar la realización.

4.1.2.3 Características de los Usuarios

El objetivo de esta tarea identificó a los responsables y/o actores de cada uno de los procesos y a los principales usuarios implicados.

Los usuarios del aplicativo fueron los siguientes:

Tipo de usuario	Administrador
Formación	Universitaria o Superior
Habilidades	Conocimiento tecnológico de la plataforma MOODLE
Actividades	Administración de la plataforma, creación, modificación o eliminación de usuario y cursos, así como configuración de la plataforma.

Tipo de usuario	Profesor
Formación	Superior o mayor
Habilidades	Uso de herramientas ofimáticas e internet
Actividades	Uso de la plataforma con perfil de Profesor, según las opciones que hayan sido dadas por el administrador.

Tipo de usuario	Alumno
Formación	Secundaria
Habilidades	Uso de herramientas ofimáticas e internet
Actividades	Uso de la plataforma con perfil de Alumno, según las opciones que hayan sido dadas por el administrador.

4.1.2.4 Restricciones

No hubo ninguna, ya que todos los usuarios brindaron toda su colaboración para obtener toda la información necesaria para llevar a cabo el desarrollo de este proyecto.

4.1.3 REQUISITOS ESPECÍFICOS

4.1.3.1 Requisitos de interfaces

4.1.3.1.1 Creación de Cursos

Requisitos Funcionales

- ✓ Una vez instalado el sistema se procedió con la creación de los cursos en el mismo.
- ✓ El Coordinador de cada área entregó al administrador del sistema la lista de cursos que se crearán.
- ✓ Junto a la lista de cursos estuvo adjunto la lista de estudiantes, el profesor que va a dictar el curso y los horarios del mismo.
- ✓ Una vez aprobados estos cursos por cada coordinador, el administrador procedió a crearlos.
- ✓ En la creación del curso asignó, profesor estudiantes, modelo de presentación y permisos de los usuarios.
- ✓ Una vez creado el curso en MOODLE el profesor asignado al mismo preparó el calendario base del mismo, y lo habilitó para el uso de los estudiantes.

4.1.3.1.2 Carga y revisión de Materia

Requisitos Funcionales

- ✓ Este proceso lo realizó cada uno de los profesores en los cursos que tiene designados para dictar materia.

- ✓ Para que el profesor pueda cargar información a un curso, el administrador previamente asignó al usuario como profesor en los cursos a los cuales estaba fijado.
- ✓ Una vez que el profesor ingresó al sistema, escogió el curso o cursos a los cuales desea cargar materia.
- ✓ La carga de materia se la realizó desde la pantalla del curso asignado.
- ✓ Una vez cargada la materia todos los estudiantes, matriculados en el mismo tuvieron la posibilidad de descargar el material.
- ✓ De requerirse una respuesta al material, los estudiantes también consiguieron subir su archivo o deber de respuesta.

4.1.3.1.3 Uso de Herramientas compartidas

Requisitos Funcionales

- ✓ Las herramientas usadas compartidas entre todos los usuarios sin tomar en cuenta su tipo (administrador, profesor, alumno, entre otros), debieron ser instaladas por el administrador en el aplicativo.
- ✓ Estas herramientas pueden ser : blog, chat, foros generales entre otros.
- ✓ Una vez instaladas las herramientas por el administrador cualquier usuario registrado del sistema pudo hacer uso de las mismas.
- ✓ Dependiendo de la herramienta que escoja el usuario podrá hacer uso de ella, y comunicarse con otro usuario o un grupo de usuarios.

4.1.3.2 Requisitos no Funcionales

4.1.3.2.1 Rendimiento

La Información almacenada por los usuarios fue actualizada y consultada en cualquier momento, tomando en cuenta que es un sistema web todo requerimiento guardado o cambio en los datos será reflejado en tiempo real para todos los diferentes tipos de usuarios.

4.1.3.2.2 Seguridad

El acceso al sistema para todo usuario pasó por un login de clave segura la misma que por seguridad tuvo un mínimo de 8 caracteres, una letra mayúscula, un número y un signo especial para ser aceptada, estos fueron encriptados en su guardado en la base de datos, para seguridad de todos los usuarios.

4.1.3.2.3 Disponibilidad

El sistema está disponible durante el rango de tiempo de 24/7/365, puesto a que trabaja en un ambiente web al cual se puede tener acceso en cualquier momento desde cualquier lugar donde exista acceso a internet y un navegador sin necesidad de ninguna instalación.

4.1.3.2.4 Mantenibilidad

La implantación de la misma está dada para que su mantenimiento sea fácil de realizarlo, hablando de un mantenimiento preventivo, para un mantenimiento correctivo se generó un plan de respaldos diarios para que en caso de daño cargar el respaldo del día anterior y así no perder información y poder continuar con el uso del sistema sin ningún inconveniente.

4.1.3.3 Requisitos Funcionales

4.1.3.3.1 Diagramas Generales

4.1.3.3.1.1 Acceso y salida del Sistema

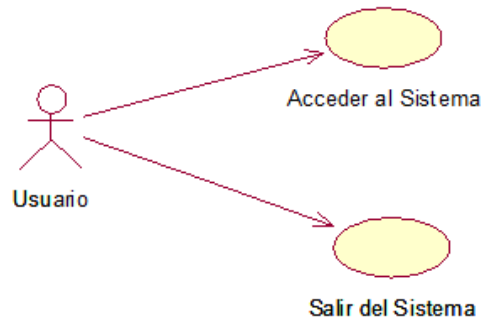


Figura 7. Acceso y Salida del sistema

Tabla 2. Especificación de CU: Acceso y Salida del sistema

Especificación del caso de uso			
Nombre del caso de Uso:	Acceso y salida del sistema	Versión No:	1.0
Descripción:	Permite al Actor acceder al sistema con el uso de sus datos de Usuario y contraseña.		
Autor:	Adrián Cisneros	Fecha:	14/12/2011
Usuario/Actor:	Usuario General		

Continuación

Flujo Básico		
Paso	Actor	Sistema
1	El usuario en cualquier explorador de internet ingresa la URL del sitio.	El Sitio mostró la primera pantalla para permitir el acceso
2	El usuario en esta pantalla ingresa su usuario y clave.	El sistema validó los datos y permitirá o denegó el acceso.
3	El usuario accede al sistema	El sistema indicó la pantalla principal según el tipo de perfil que tenga el usuario.

4.1.3.3.1.2 Casos de uso del Administrador

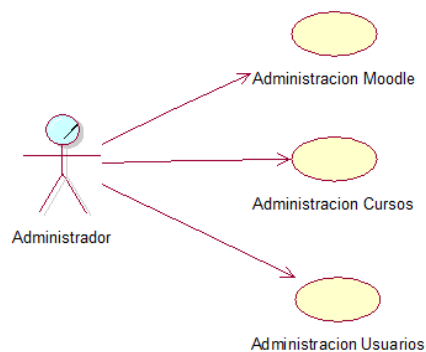


Figura 8. Casos de Uso del Administrador

Tabla 3. Descripción de CU del Administrador

CASO DE USO	DESCRIPCIÓN
Administración Moodle	En este proceso el único actor es el administrador ya que él se encarga de dar estilo y configuraciones avanzadas al sistema en favor de los demás perfiles.
Administración Cursos	En este Caso de uso actúan como actores el administrador el cual está encargado la creación, eliminación y modificación de niveles y cursos en el sistema, así también el administrador puede agregar o eliminar los usuarios de los mismos.
Administración Usuarios	El actor administrador crea los usuarios del sistema y asigna el tipo de usuario del mismo y de ser necesario los agrega a un curso, dando perfiles de acceso al sistema.

4.1.3.3.1.3 Casos de Uso del Profesor

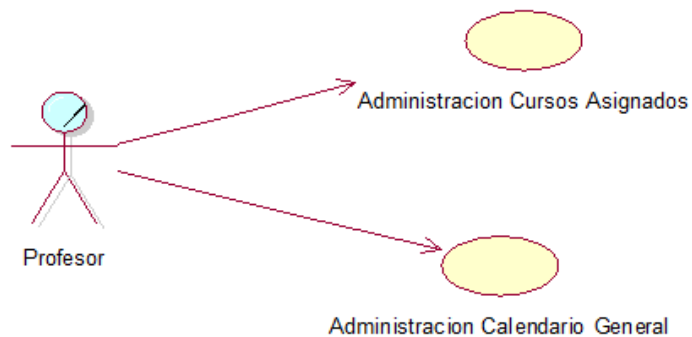


Figura 9. Casos de Uso del Profesor

Tabla 4. Descripción de Casos de Uso del Profesor

CASO DE USO		DESCRIPCIÓN
Administración Asignados	Cursos	En este proceso el profesor puede agregar y calificar trabajos, deberes y pruebas, así como ingresar las notas de sus alumnos, entre otras opciones más.
Administración General	Calendario	Aquí el profesor puede ingresar la fecha de entrega de tareas, días en los que vaya a tomar pruebas, de esta manera los estudiantes podrán ser recordados de estos eventos.

4.1.3.3.1.4 Casos de uso del Alumno

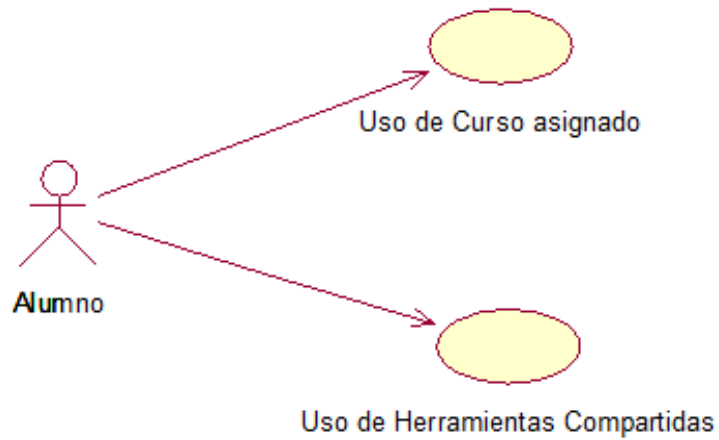


Figura 10. Casos de Uso del Alumno

Tabla 5. Descripción de casos de uso del Alumno

CASO DE USO	DESCRIPCIÓN
Uso de Cursos Asignados	El actor estudiante tiene acceso al curso al cual está asignado aquí puede realizar las actividades dadas por el profesor, y compañeros., como subir deberes, dar exámenes, usar foros entre otros.
Uso Herramientas Compartidas	El estudiante puede hacer uso de las herramientas compartidas con sus compañeros y demás usuarios del sistema.

4.1.3.3.1.5 Casos de Uso de interacción entre Profesor y Alumno

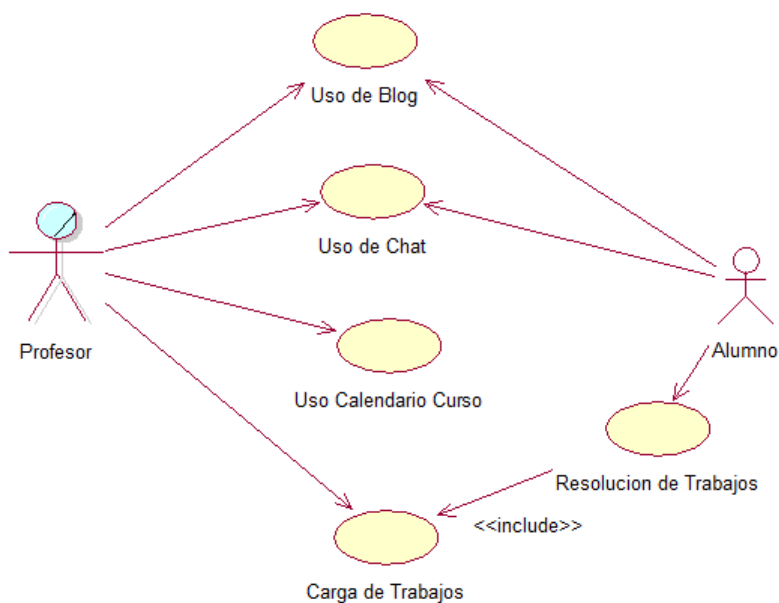


Figura 11. Caso de Uso Interacción entre Profesor y Alumno

Tabla 6. Descripción Caso de Uso Interacción entre Profesor y Alumno

CASO DE USO	DESCRIPCIÓN
Uso del Blog	En este caso de uso actúan tanto el profesor como el estudiante. Comentando el tema del blog abierto por cualquiera de los dos actores.
Uso del Chat	El chat se utiliza para entablar una conversación en línea entre los usuarios del sistema.
Uso Calendario Curso	El calendario del curso puede modificarlo solo el profesor y solo de los cursos a los que tiene acceso, esto quiere decir, añadir, eliminar, y modificar actividades que tenga con los estudiantes.
Carga de Trabajos	El actor profesor es el encargado de cargar los trabajos siendo estos, pruebas o exámenes para ser resueltos por los estudiantes.
Resolución de Trabajos	El actor alumno puede subir la resolución de sus trabajos al sistema para que puedan ser revisados por el actor profesor, este caso depende de que el profesor haya cargado un trabajo al sistema.

4.2 FASE DE ELABORACIÓN – DISEÑO

Se realizó un análisis para determinar que metodología usar, mediante el siguiente test, de preguntando a las personas que ayudaron a realizar la implementación del sistema, el resultado fue el siguiente, estas respuestas se encuentran en los anexos.

Tabla 7 Preguntas para confirmación de Metodología a utilizar

Tiempo que puede utilizar para Muy poco
trabajar en el desarrollo del sistema

Conocimiento de los procesos entre profesor y estudiante	Bastante
Manejo del Sistema LMS Moodle	Bastante
Versión utilizada de Moodle	Moodle 1.9

Tomando en cuenta estos resultados fue necesario la utilización de una metodología que permita, utilizar de gran manera los conocimientos de los futuros usuarios en el poco tiempo que ellos pueden aportar para la implementación del proyecto, por ello se utilizó la metodología XP la misma que cumplió con todo los requisitos necesarios para la realización del proyecto en el menor tiempo posible, esta metodología ayuda a planificar el producto final utilizando en una primera cita el conocimiento de los futuros usuarios, teniendo esto como la primera gran base para la creación y parametrización del mismo, después se realizaran reuniones para cambios menores en el sistema.

Las diferentes etapas que están definidas en el punto 2.2.3.8 del Marco Teórico, son las siguientes:

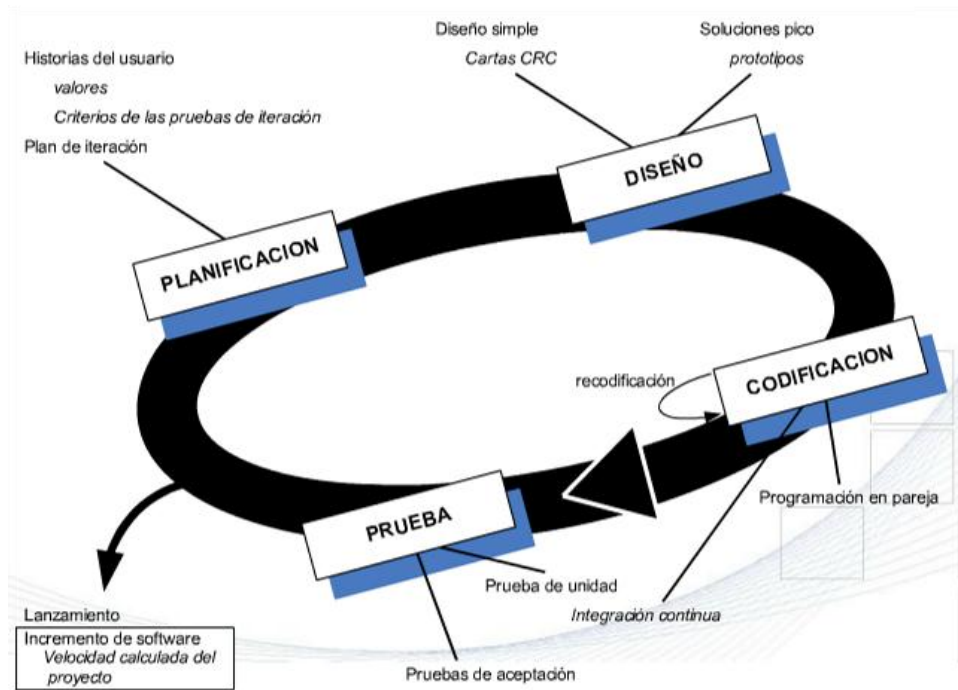


Figura 12. Fases Metodología XP

4.2.1 PLANIFICACIÓN

En donde se demostró la factibilidad y tiempo de la implementación del sistema MOODLE en la sección nocturna del Colegio Intisana de Quito

4.2.1.1 FACTIBILIDAD

Se refiere a la disponibilidad de los recursos necesarios para llevar a cabo los objetivos señalados, es una de las primeras etapas del desarrollo de un sistema informático.

La factibilidad del uso de moodle en cualquier institución se da por su simpleza en el momento de parametrizar, esto permite que se adapte fácilmente a los procesos que las instituciones tienen, es por eso que moodle es uno de los primeros LMS utilizados no solo al nivel de colegio si no de Universidades e institutos educativos en general; a continuación se resuelve factibilidad técnica, económica, operativa y legal, para reconfirmar la aplicación de MOODLE en el Colegio Intisana de Quito

Para obtener la idea principal del proyecto. Se realizó un análisis que permitió diagnosticar la problemática a la que se dio solución. Usando una metodología de estudio de la situación interna y externa competitiva del proyecto de tesis en sus efectos para determinar sus Fortalezas, Oportunidades, Debilidades y Amenazas.

La situación interna se compone de dos factores controlables: fortalezas y debilidades, mientras que la situación externa se compone de dos factores no controlables: oportunidades y amenazas.

A continuación, se detalla un análisis FODA, el cual sirvió para emitir un diagnóstico sobre si existe o no la necesidad de desarrollar el proyecto.

Tabla 8. Análisis FODA

Análisis FODA	Fortalezas	Debilidades
Análisis Interno	Buen grupo de trabajo. Contar con recursos económicos. Tener una base de conocimientos de informática. Mantener la información de profesores y alumnos al día.	Resistencia de los profesores al cambio. Alumnos que no tienen un acceso fácil al uso del Internet.
	Oportunidades	Amenazas
Análisis Externo	Mejor posicionamiento como colegio, en uso de nuevas tecnologías. Política crediticia favorable. Tecnología al alcance de su mano.	Reformas en la ley del Educación del Ecuador. Costo en el uso del sistema tanto en licencias como en uso del servidor Que el sistema moodle se vuelva obsoleto.

De la combinación de fortalezas y oportunidades surgen las potencialidades, las cuales señalan las líneas de acción más prometedoras para el proyecto de sistema MOODLE.

Las limitaciones, surgen de las debilidades y amenazas, colocan una seria advertencia.

Los riesgos (combinación de fortalezas y amenazas) y los desafíos (combinación de debilidades y oportunidades), determinados por su correspondiente combinación de factores, exigirán una cuidadosa consideración a la hora de marcar el rumbo que la organización deberá asumir hacia el futuro deseable.

De acuerdo al FODA realizado inicialmente, se pronosticó los beneficios del MOODLE y lo que sucedería si no se establecen cada una de las amenazas y debilidades del proyecto en su aplicación.

Examinando las debilidades como la resistencia de los profesores al cambio ya que en estos tiempos la utilización de la informática es muy común, y su implementación para algunos profesores en el método de enseñanza con la utilización de la tecnología podría ocasionar un poco de molestia al no estar acostumbrados a estos procesos.

Cuando los alumnos que no tienen un acceso fácil al uso del Internet puesto a que los alumnos de la sección nocturna no disponen de recursos económicos, esto puede generar un problema, en el momento que no logren acceder al sistema para la realización de sus actividades académicas diarias.

Considerando las amenazas, como las reformas en la ley del educación del Ecuador, en el caso de una reforma que impida el uso de estas plataformas para el caso de llevar calificaciones o información de los estudiantes de la institución.

Costo en el uso del sistema tanto en licencias como en uso del servidor, el sistema para mantenerse en el internet esta instalado en un servidor dedicado y tiene un costo anual su host en internet, el precio del host puede variar aunque no en gran manera, pero si la institución sigue creciendo de igual manera lo hará el número de usuarios y la información que ingrese al moodle llegará a un punto que necesite un servidor con más capacidad de procesamiento y de

almacenamiento lo cual significaría un costo alto para la mantención del sistema, además que existe la posibilidad que el uso de moodle deje de ser libre a tener un coste de hacer esto realidad se tendría que hacer un balance entre la necesidad del sistema y si es justificable su mantención.

Que el sistema MOODLE se vuelva obsoleto, siendo una herramienta libre solo realiza actualizaciones si usuarios siguen creando nuevos módulos y así generando nuevas versiones, si se interrumpe esta generación de versiones MOODLE correría el riesgo de volverse obsoleto a nuevas tecnologías. Por el momento los expertos dan a MOODLE por lo menos 5 años para no convertirse en una plataforma obsoleta.

Se debe realizar el control del pronóstico, el cual hará posible el seguimiento y disminución de los problemas mencionados.

Automatizar la interacción alumno-profesor y la retroalimentación de la asignatura permitirá al Colegio:

Adecuar un gestor en línea capaz de promover un aprendizaje más eficaz y abaratar costos, precisar de una herramienta que permita gestionar los diferentes procesos que se circunscriben en el colegio así como la interacción entre profesores y alumnos.

Tener una base de conocimientos de informática con la información de actividades entre profesores y alumnos.

De ser posible adquirir un servidor para mantener históricos de la información para no generar la necesidad de comprar un servidor con más capacidad para el trabajo y uso actual, mantener siempre el Moodle con las últimas actualizaciones teniendo en cuenta los módulos más importantes con los que la institución trabaja.

4.2.1.1.1 FACTIBILIDAD TÉCNICA

La tecnología moderna, permite realizar sistemas informáticos muy complejos que agilitan y optimizan todo tipo de procesos. El presente proyecto de tesis fue factible de realizar porque las herramientas, equipos tecnológicos y software necesarios se encuentran disponibles en el mercado.

Se puede ver en el siguiente análisis, los requerimientos mínimos que son necesarios para instalar MOODLE en un servidor ya que el cliente solo necesita acceso a internet mediante un navegador web, el cliente puede utilizar así una tableta, una PC o Laptop para poder utilizar MOODLE.

Se tomó en cuenta la combinación del software y del hardware utilizado para el tipo de uso que va a tener; el uso concurrente de varios usuarios está contemplado en la memoria y el procesador que tiene el servidor.

Software

En reunión con las autoridades del Colegio se descartó la posibilidad de implementar el sistema en los servidores de ellos, ya que sería necesario la adquisición de uno nuevo y la compra de una IP pública para la salida a internet del sitio, la compra de licencias y más, por lo cual se tomó la decisión de adquirir un servidor dedicado en una ISP (*Internet Service Provider*), lo cual económicamente resultaba más conveniente para la institución.

La siguiente tabla muestra el software que es recomendado por la página de MOODLE y el software que fue utilizado para la configuración del Servidor, donde se encuentra instalado el sistema.

Tabla 9. Factibilidad técnica de software

Software			
Software	Recomendado	Utilizado	Razón de uso
<i>Sistema Operativo</i>	Windows XP/2000/2003 o Linux	Linux	Trabaja de mejor manera con Moodle
<i>Servidor Web</i>	Apache o IIS	Apache	Permite una excelente configuración de archivos PHP lo cual hace que moodle se mantenga estable
<i>PHP</i>	Versión 5.3.2 en adelante	Versión 5.3.2	Es la versión probada y que trabaja de mejor manera con moodle
<i>Base de Datos</i>	MySql o PostgreSql	MySql 5.0.25	Tiene bastante documentación y soporte en el internet además de haber sido probada como la base mas estable q trabaje con Moodle

Hardware

El Servidor se arrienda junto con el host en el ips, éste es un servidor dedicado es decir que solo se lo utiliza para la institución. Se tomó la decisión de alquilar el servidor por cuestiones económicas ya que adquirir un propio servidor es muy costoso.

A continuación la siguiente tabla muestra las características recomendadas y las que tiene el servidor rentado.

Tabla 10. Factibilidad técnica de hardware

Hardware			
Hardware	Recomendado	Utilizado	Razón de uso
<i>Espacio de Disco</i>	Mínimo de 160Mb de espacio libre, 5 Gb mínimo total	2 Tb	Ya que el sistema es utilizado por toda la institución es necesario tener gran capacidad de almacenamiento para toda la carga de información q se requiere
<i>Respaldos</i>	Semanalmente	Semanalmente	Los respaldos se llevan en otra computadora para no afectar el espacio en disco del servidor, este computador es propio del colegio.
<i>Memoria</i>	Mínimo de 250Mb, 1 Gb es una mejor recomendación	Versión 5.3.2	El servidor tiene 16 GB de memoria esto permite que su procesamiento sea más rápido y no colapse con el ingreso de varios usuarios al mismo tiempo

4.2.1.1.2 FACTIBILIDAD ECONÓMICA

Económicamente el diseño del sistema e Implantación en un futuro si es factible, considerando que MOODLE así como todas las herramientas de software como PHP, MySQL, Linux Red Hat, son software libre, es decir no tienen costo de su utilización.

Los costos incluirían, la compra de dominios que se deseen en esta implementación, el Colegio contará con dos dominios, el host o alojamiento en un servidor dedicado que pueda abarcar toda la información del Colegio, y un

mantenimiento trimestral de la plataforma. La siguiente tabla detalla estos costos.

Tabla 11. Análisis de Costos

Descripción	Valor por hora (\$)	Trabajo en horas	Costo total (\$)
Costo anual del servidor dedicado			\$380.00
Horas Internet	\$0.50	640	\$320.00
Horas Maquina	\$0.30	640	\$192.00
Horas Implantación	\$3.00	640	\$1,920.00
Costo Dominios		3	\$60
TOTAL			\$2,872.00

Se tomó en cuenta que el costo del host va bajando acorde los años que se lo tenga activo, inclusive existen promociones por la compra de varios años.

En un análisis costo beneficio acorde al costo de 2,872.00 dólares americanos, se determina que con este costo no solo la sección nocturna puede tener acceso a la plataforma si no también se puede incluir la sección diurna del colegio, con esto sería beneficiada toda la institución con este coste inicial relativamente bajo para el número de estudiantes y profesores pertenecientes a la misma que harán uso del sistema.

También estar al tanto que este costo es inicial y que en caso de necesitar un aumento de capacidad el pago anual del servidor contratado no crece en gran manera, y en cuanto al mantenimiento de la plataforma esta no requiere de más

que un gasto trimestral, de igual manera relativamente bajo, comparado con la ayuda y facilidades que da MOODLE a todos los usuarios que lo utilicen.

Poniendo como ejemplo el uso de papel puesto que MOODLE permite la carga de archivos los trabajos y deberes podrán ser entregados de esta manera así como reporte de notas entre otros, con esto se elimina en gran manera el gasto en papel y utilitarios físicos para el estudio.

El tiempo también es un factor a considerar y en el cual MOODLE es de gran ayuda ya que al no necesitar la presencia física del profesor o del estudiante y permitir el envío de asignaciones la carga de ellos, comentar en foros o blogs temas de clases permite que el tiempo presencial en clase sea aprovechado de mejor manera.

4.2.1.1.3 FACTIBILIDAD OPERATIVA

La administración del software final y del equipo tecnológico es operable por alumnos, profesores y administradores de la plataforma.

Función Operativa del Usuario en el Software.

Ya que moodle se centra totalmente en ser una herramienta fácil de utilizar cada uno de sus actores podrá guiarse fácilmente por sus pantallas para realizar las tareas que desee.

Por ejemplo el usuario alumno, ingresando con su usuario y contraseña se encontrará con la pantalla de los cursos a los que está matriculado y si está habilitado podrá ver si puede matricularse en otra materia, de aquí en adelante el sistema aparte de ser intuitivo cuenta con mucha ayuda de su uso en el internet.

El perfil de profesor de igual manera que el del alumno cuenta con su clave y usuario de esta manera al ingresar podrá ver los cursos que tiene asignado y

dentro de cada curso los alumnos que están matriculados en el mismo, tiene también la posibilidad del envío de trabajos a un curso designado a todos los cursos a un grupo de estudiantes que él escoja, esto de manera muy intuitiva a ayudándose en el manual de usuario.

El perfil de administrador es el perfil en el cual se necesita que el usuario tenga mayor experiencia en el uso de un ordenador ya que este estará a cargo de parametrizar las opciones del sistema en general y las opciones de cada uno del grupo de usuarios, también tiene en sus opciones la creación y eliminación de usuarios en el sistema.

Para contar con un mejor uso del sistema se capacitó a un grupo de usuarios de cada tipo, es decir: profesor y estudiante, además de un súper usuario el cual será el administrador del sistema.

4.2.1.1.4 FACTIBILIDAD LEGAL

El instituto tiene un modelo de notas trimestral, cada una de estas notas es generada por tres notas en el trimestre dadas por deberes, trabajos y pruebas.

Acorde al reglamento interno de la institución cada nota trimestral está dada sobre una base de 10 puntos, como máximo, el puntaje mínimo que el estudiante debe tener es de 7 puntos cada trimestre.

Apoyando a este sistema MOODLE permite el envío la carga y recepción de trabajos y deberes, además de trabajar con una fuerte herramienta que permite personalizar los exámenes agregando las preguntas y respuestas dando así la posibilidad de una auto calificación para que el estudiante pueda ver su nota después de terminar el examen y que esta nota sea agregada de inmediato al registro de notas del alumno en esa materia, y al record del profesor.

Aplicando esto es muy sencillo llevar el registro de notas de los estudiantes y como la información se la puede imprimir de esta manera no tiene un fuerte

impacto en la forma de llevar registros de calificaciones al colegio, si desea llevarlos impresos o en digital como actualmente lo hacen de igual manera solo se hará una exportación de la información a Excel.

4.2.2 DISEÑO

MOODLE, se refiere a "objetos de aprendizaje", es decir tiene un diseño de objetos normalmente diseñados de tamaño pequeño para distribuirse en internet posibilitando el acceso simultáneo a la información por parte de múltiples usuarios.

Este hecho es fundamental ya que a partir de ahora no se basan en la simple lectura de unos apuntes sino en la creación de estos "objetos de aprendizaje", plenos de significado, que siguen secuencias didácticas en las que el profesor guía a los alumnos posibilitando su auto aprendizaje. Facilitando así el aprendizaje individual y la colaboración entre los participantes. El objetivo sería crear unidades didácticas que responderían a las diferentes capacidades a desarrollar en la asignatura.

4.2.2.1 DISEÑO DE INTERFACES DE USUARIO

MOODLE se caracteriza por contar con interfaz gráfica que permite a los usuarios identificar fácilmente las opciones de navegación que cada uno tiene según su perfil, ahorrando así tiempo en navegación.

MOODLE permite descargar varios temas en la siguiente dirección, <http://moodle.org/mod/data/view.php?d=26>, también da la opción de crear una propia interfaz gráfica, agregando textos, imágenes, logos e inclusive modificar la posición de menús e iconos de ingreso a las opciones.

A continuación se observa las pantallas de inicio que se utilizaron para algunos tipos de usuarios.

Pantalla del Administrador

La pantalla muestra el inicio del administrador, este perfil como se muestra en el árbol de ingreso de la izquierda tiene la posibilidad de administrar usuarios, cursos, el sistema en general en la parte media tiene todas las secciones y cursos y un calendario en la parte izquierda que muestra las actividades generales del colegio.

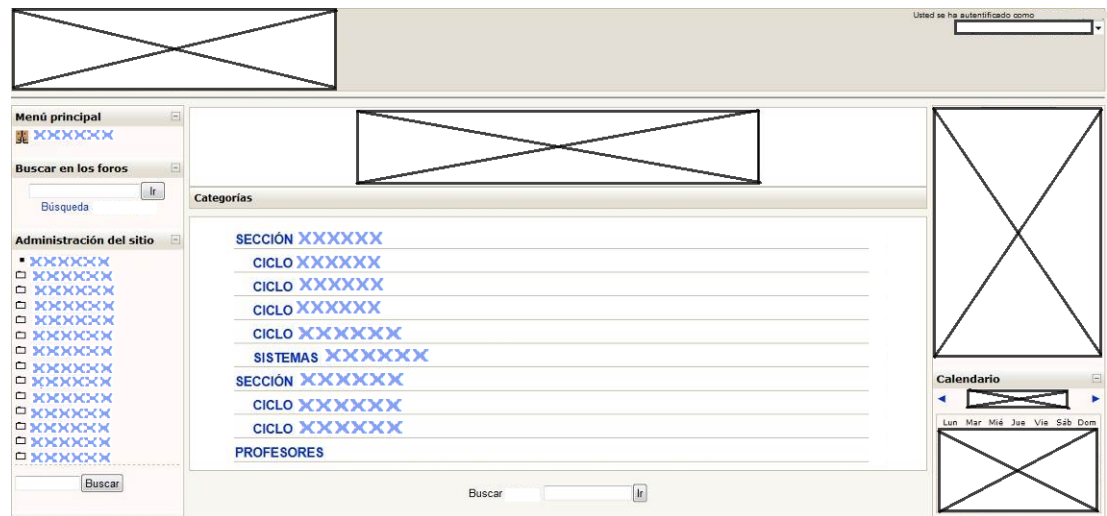


Figura 13. Pantalla Base Administrador

Pantalla Estudiante

Esta pantalla demuestra cuando el estudiante ingresa a uno de los cursos que se encuentre matriculado, en la parte izquierda tiene acceso directo a actividades calificaciones y foros, en la parte media se identifica novedades foros o blogs del curso o generales, y en la parte derecha tiene el calendario para poder organizar y verificar los eventos académicos que tenga como: pruebas, entrega de trabajos, entre otros.

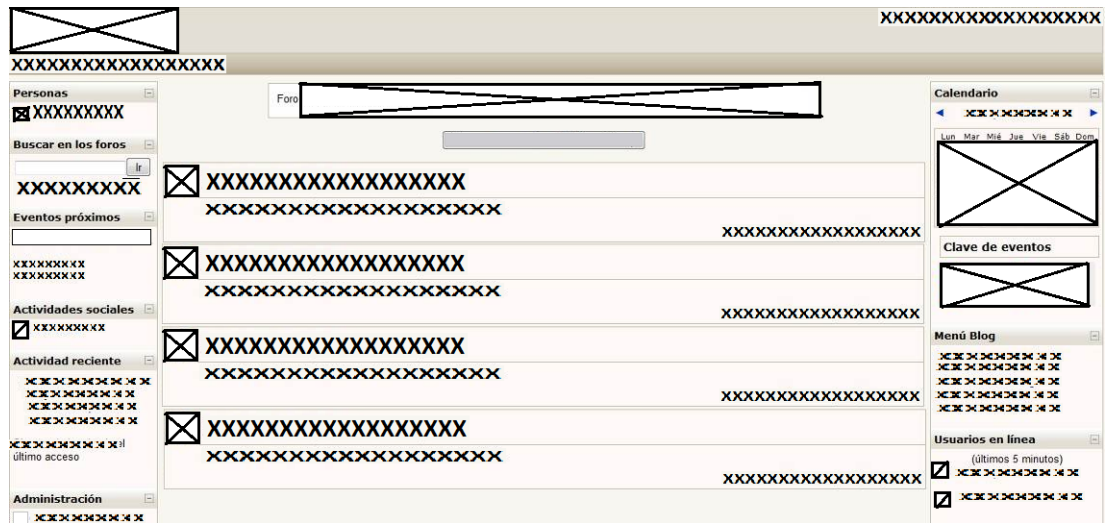


Figura 14. Pantalla Base Estudiante

Pantalla Profesor

La pantalla a continuación es la primera que ve un profesor al ingresar a su perfil, la cual muestra los cursos en los cuales dicta clases, y a los cuales puede ingresar y administrar.



Figura 15. Pantalla Cursos Profesor

La figura siguiente es la pantalla en la que ingresa el profesor una vez escogida la materia que desea administrar, en la parte izquierda tiene la posibilidad de manejar este curso como: crear un grupo de sus estudiantes, asignar trabajos, pruebas, así también poner las calificaciones de los mismos.

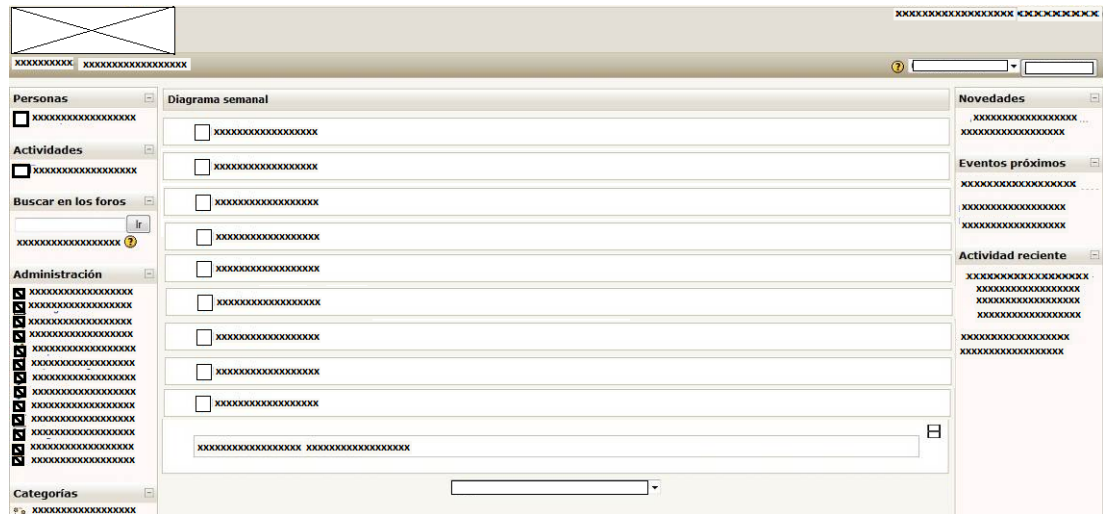


Figura 16. Pantalla Base Administración Curso Profesor

4.2.2.2 DISEÑO DE LA BASE DE DATOS

Tomando en cuenta el objetivo que de la creación de MOODLE se generó su base de datos de manera que cumpla con todos los objetivos para los cuales ha sido creado. Aquí se presenta el modelo entidad relación de la Base de Datos y en la parte de anexos se encontrará el modelo conceptual detallando cada una de las tablas con las que trabaja MOODLE.

4.2.3 CODIFICACIÓN

4.2.3.1 CREACIÓN DE SECCIONES DEL COLEGIO INTISANA

Para la generación de las secciones se debe crear primero los cursos para esto se siguen los siguientes pasos.

Lo primero es tener un perfil de administrador, sólo éste tiene la facultad de crear o eliminar cursos y grupo de cursos.

En el caso del Colegio Intisana hay dos grupos grandes los cursos de sección Diurna y el otro los cursos de sección Nocturna.

Los siguientes pasos:

Primero.-se dirigen a la administración del sitio en el árbol de opciones se identifica la carpeta Cursos, en la cual se escoge Agregar / editar curso.



Figura 18. Primer paso para agregar nuevo Curso

El siguiente paso depende de lo que desee realizar, el botón Agregar un nuevo curso, lleva a la pantalla en la cual genera un nuevo curso en la raíz o dentro de una categoría.

El botón Agregar nueva categoría nos permite crear un nuevo grupo en donde se guardan los cursos pertenecientes a tal categoría, por ejemplo la categoría Cuartos Cursos, y los cursos de ellos podrían ser Cuarto Sociales, Cuarto Matemática.

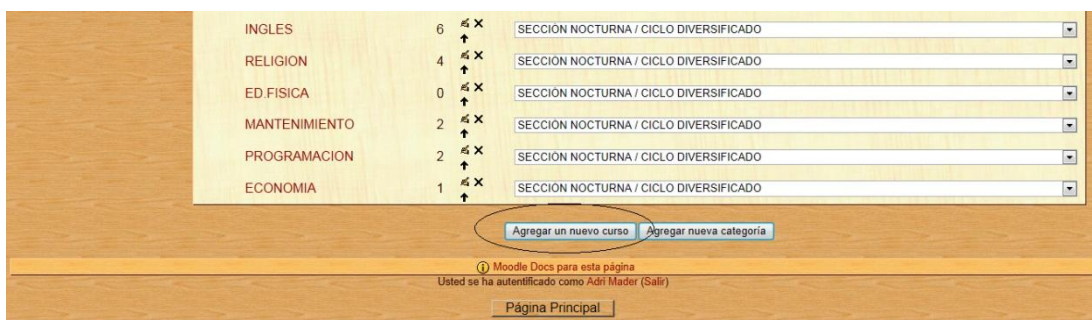





Figura 19. Segundo paso para agregar nuevo Curso


En el siguiente paso aparece la siguiente pantalla en la cual hay que llenar la información necesaria en cada uno de ellos, se debe tener en cuenta que los campos en rojo con asterisco son obligatorios. Si existe alguna duda de que información va en algún casillero se puede dar clic en el icono de ayuda a un lado del nombre del casillero, allí se desplegará una ayuda que indicará que tipo de información es necesaria y para qué sirve la misma.


Ajustes generales



















Categoría  SECCIÓN NOCTURNA

Nombre completo*  Nombre del curso


Nombre corto*  ID del curso


Número ID del curso 


Resumen 


Trebuchet | 1 (8 pt) | Idioma | **B** | **I** | **U** | **S** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 


Ruta:


Formato  Formato semanal


Número de semanas o temas  10

Fecha de inicio del curso  20 | abril | 2012

Temas ocultos  Las secciones ocultas se muestran en forma colapsada

Items de noticias para ver  5

Mostrar calificaciones  Si

Mostrar informes de actividad  No


Tamaño máximo para archivos cargados por usuarios  10Mb

Figura 20. Tercer paso para creación de Cursos

Para finalizar se da clic en guardar cambios y el curso está listo para ser usado por los usuarios a los que se les agregue, tanto profesor como estudiantes.

Grupos

Modo de grupo  No hay grupos

Forzar  No

Disponibilidad

Disponibilidad  Este curso está disponible para los estudiantes

Contraseña de acceso  Desenmascarar

Acceso de invitados  No admitir invitados

Idioma

Forzar idioma

Renombrar rol 

Administrator

Course creator

Teacher

Non-editing teacher

Student

Guest

Authenticated user

tutor

Figura 21. Cuarto paso para creación de Cursos

4.2.3.2 CREACIÓN DE USUARIOS

La creación de usuarios se la puede realizar de dos maneras, la una creación de usuario por usuario individualmente, y la otra creando archivos de carga masiva.

Para el caso de la institución, se decidió la creación de usuarios en carga masiva con el fin de superar el alcance y crear la mayor cantidad de usuarios en el menor tiempo posible.

Los pasos para la creación de usuarios utilizando carga masiva se la realiza de la siguiente manera:

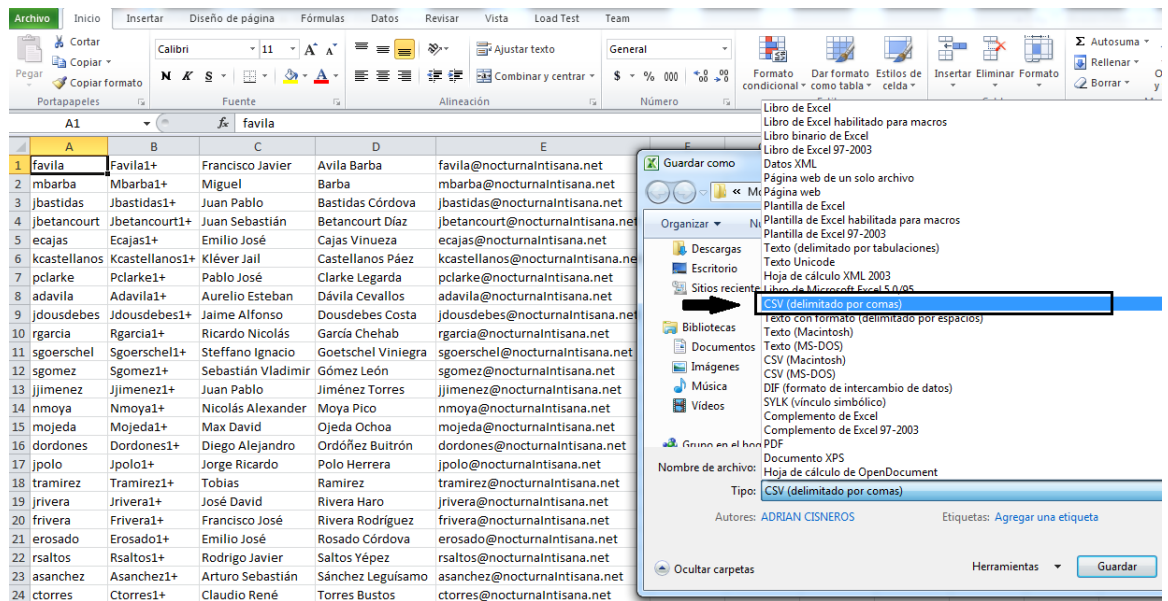


Figura 22. Creación de Archivo CSV para carga masiva

El primer paso es crear un archivo plano en Excel. Como se puede ver en la figura 21 el orden que se debe de tener es el siguiente:

Columna A.....Usuario

Columna B.....Clave

Columna C.....Nombres

Columna D.....Apellidos

Columna E.....Mail

Hay que tener en consideración que en los campos de nombre un apellido solo acepta letras y no deben tener tildes, la clave debe contener al menos una letra mayúscula, un número y un signo especial, con esto la creación de usuarios no tendrá problemas.

Una vez ingresada la información en el archivo de Excel el siguiente paso es: ir a archivo guardar como y escoger en tipo, CSV delimitado por comas como se pudo ver en la figura 21.

Lo siguiente es ir al MOODLE e ingresar como un usuario administrador y seguir los siguientes pasos.

1.- En el árbol principal de opciones dar clic en Cuentas y dentro de esta carpeta en Subir Usuarios



Figura 23. Pantalla Para subir Usuarios Masivamente

Una vez que estemos en la pantalla para subir usuarios masivamente (Figura 22) damos clic en examinar y buscamos nuestro archivo antes creado (Figura 21), después se escoge como limitador la, (Punto 4; Figura 22) en codificación se selecciona UTF-8 (punto 5; Figura 22), después en pre visualizar filas (punto 6; Figura 22) se escoge las que nosotros deseemos y finalmente damos clic en Subir usuarios (punto 7; Figura 22).

Una vez terminado este proceso nos mostrará la lista de usuarios que han sido creados satisfactoriamente.

4.2.3.3 IMPLANTACIÓN

En la implantación del sistema se realizó la instalación del servidor con un sistema Linux, un servidor Apache y MOODLE en su versión 1.9, todo esto bajo los estándares vistos en el punto de factibilidad técnica.

Las direcciones a las cuales se pudo acceder al sitio web son: <http://www.nocturnaintisana.net/> o <http://www.intisana.org/>, la pantalla a la cual se accede en estas direcciones es la siguiente:



Figura 24. Pantalla Principal de Ingreso

Aquí los usuarios pudieron ver la información general de la plataforma, el calendario general donde se ven las actividades públicas del Colegio, novedades entre otra información, y la parte de ingreso de usuario y contraseña para validación de su perfil en el sistema.

El Ing. Robert Montero Administrador del sistema, fue capacitado para tener todos los conocimientos de cómo administrar la plataforma y guardar su información en caso de fallo de software o hardware.

4.2.4 PRUEBAS

Para la etapa de pruebas, se realizó con los tres tipos de usuarios; se obtuvieron resultados satisfactorios ya que el sistema funcionó de la forma adecuada de acuerdo a las siguientes tablas de satisfacción.

Y la imagen de utilización de usuarios después de una semana de uso, además de que los usuarios ya no requirieron cambios en la plataforma.



Figura 25. Ingreso de Usuarios en sistema Pruebas Satisfactorias

Como se ve en la figura 24 el ingreso de varios usuarios al sistema no afectó de ninguna manera su desempeño, estos usuarios ingresaron indistintamente a varios cursos de las secciones creadas, con esto el sistema brindó la seguridad que está listo para su utilización en casos reales.

Haciendo uso del Programa Web Server Tool 7 se hicieron pruebas de rigor las cuales se podrá ver el soporte real que tiene la página al ingreso de muchos usuarios simultáneamente, carga de información, clics en un mismo tiempo, carga de cookies, entre otros.

Muchos de las aplicaciones web corren muy bien siempre que las utiliza con una cantidad mínima de usuarios utilizándola al mismo tiempo, pero que pasa el momento en que cientos de usuarios acceden al sitio simultáneamente, usando la aplicación Web Server Stress Tool se realizó la simulación de varios patrones de ingreso con carga masiva dentro del sitio <http://intisana.org> para comprobar cuanta carga masiva simultanea es capaz de soportar tomando en cuenta que

la aplicación ahora es utilizada por la sección nocturna y diurna del Colegio Intisana de Quito.

4.2.4.1 RESULTADOS

Una vez pasada las pruebas con el programa Web Stress Tool 7 para la aplicación MOODLE estos fueron los resultados. En ambas pruebas se utilizaron las siguientes urls pertenecientes al sistema MOODLE.

Tabla 12. Urls usadas para pruebas de rigor

URL#	Name	Click Delay [s]	URL	Username	password
1	INICIO	5	http://intisana.org/		
2	LOGIN	5	http://intisana.org/login/index.php	admin	*****
3	CURSO	5	http://intisana.org/course/edit.php		
4	ROLES	5	http://intisana.org/admin/roles/sign.php?contextid=3		
5	CAT	5	http://intisana.org/course/category.php?id=1&sesskey=4RFQ5hu3W7&categoryedit=1		

En la prueba de time, o también conocida como “burn in test”, es la prueba que dedujo cuanta carga puede soportar el servidor, los resultados fueron los siguientes.

Tabla 13. Configuración para la prueba time

Project and Scenario Comments, Operator

Test Setup	
Test Type:	TIME (run test for 10 minutes)
User	10 simultaneous users - 5 seconds between clicks (Random)
Simulation:	
Logging	Log every 5 seconds
Period:	
URLs	
URL	Users visit first 1 URL(s), then random URLs, then last 0
Sequencing	
:	
Browser Settings	
Browser	User Agent: Mozilla/5.0 (compatible; Webserver Stress Tool 7; Windows)
Simulation:	
Browser	Simulate Maximum Data Rate: 120 kbit/s
Simulation:	
Browser	Enable Cookies
Simulation:	
Recursive	Download Images
Browsing /	
HTML	
Parsing:	
Recursive	Download EMBED, OBJECT, FLASH
Browsing /	
HTML	
Parsing:	
Options	
Advanced	Perform Link Check
Settings:	
Logging:	Write detailed log(s)
Logging:	Write HTML data to log
Local IPs:	URL#5: GET
Automatic	http://intisana.org/course/category.php?id=1&sesskey=4RFQ5hu3W7&categorye dit=1 POSTDATA= Click Delay=5
Client System	
System	Windows NT V6.1 (Build 7601) Service Pack 1, CPU Proc. Lev. 686 (Rev. 9474) at 2133 MHz,
Memory	1193 MB available RAM of 3072 MB total physical RAM, 3445 MB available pagefile, 20434 MB free disk space on C:
Test Software	
Webserver	7.2.2.261 Trial Version
Stress	
Tool:	

Resultados por usuario.-

Tabla 14. Resultados por usuario en prueba time

User No.	Clicks	Hits	Errors	Avg. Click time [ms]	Bytes	kbit/s
1	167	295	0	1.475	2.095.268	68,48
2	173	299	0	1.489	2.180.602	67,73
3	165	292	0	1.442	2.071.296	70,09
4	172	307	0	1.507	2.170.876	66,99
5	177	312	0	1.521	2.224.279	66,47
6	170	287	0	1.510	2.127.906	66,70
7	165	287	0	1.485	2.068.981	67,95
8	166	293	0	1.446	2.082.886	69,82
9	165	294	0	1.445	2.086.310	70,01
10	174	297	0	1.402	2.177.044	71,80

Resultados por usuario.-

Tabla 15. Resultado por url en prueba time

URL No.	Name	Clicks	Errors	Errors [%]	Time Spent [ms]	Avg. Click Time [ms]
1	INICIO	10	0	0,00	22.650	2.265
2	LOGIN	462	0	0,00	656.042	1.420
3	CURSO	392	0	0,00	673.798	1.719
4	ROLES	392	0	0,00	657.951	1.678
5	CAT	413	0	0,00	456.184	1.105

Gráficos de resultados

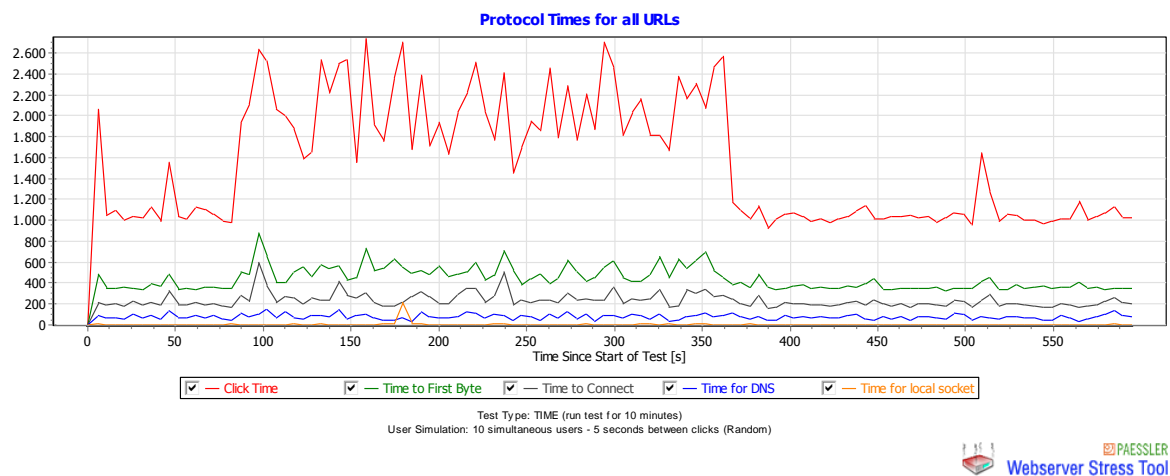


Figura 26. Tiempo de respuesta en prueba Time

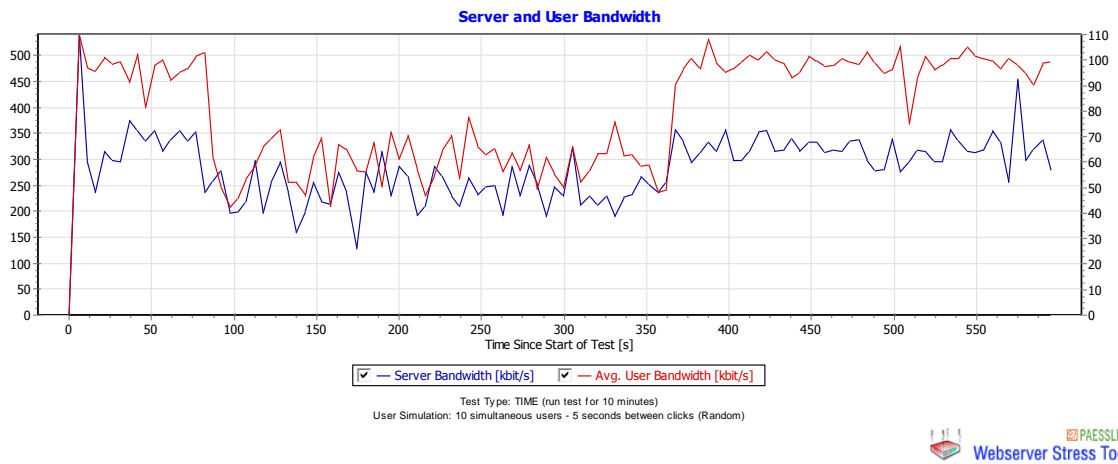


Figura 27. Uso de ancho de banda en prueba time

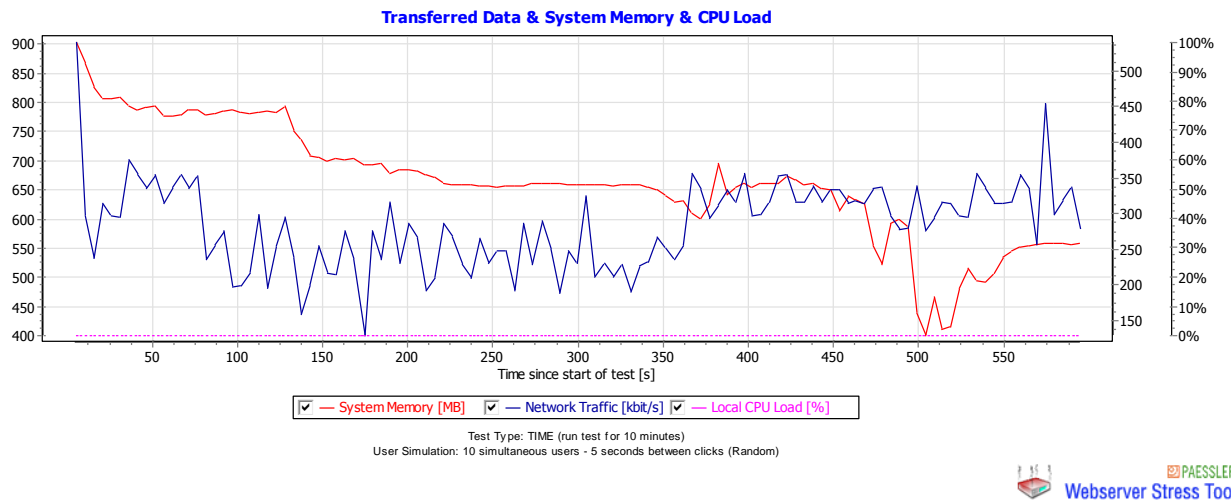


Figura 28. Uso del cliente en prueba time

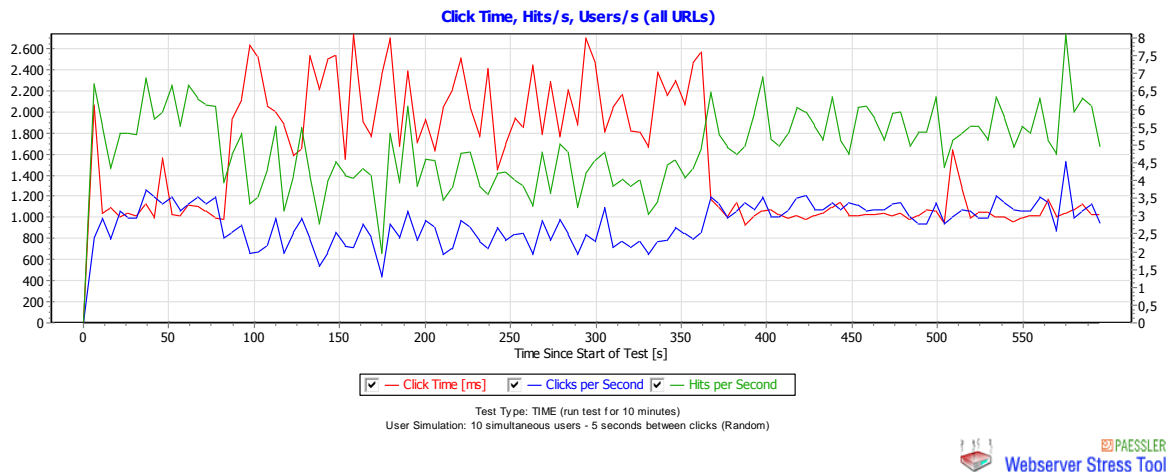


Figura 29. Promedio de espera en respuesta a requerimiento en prueba time

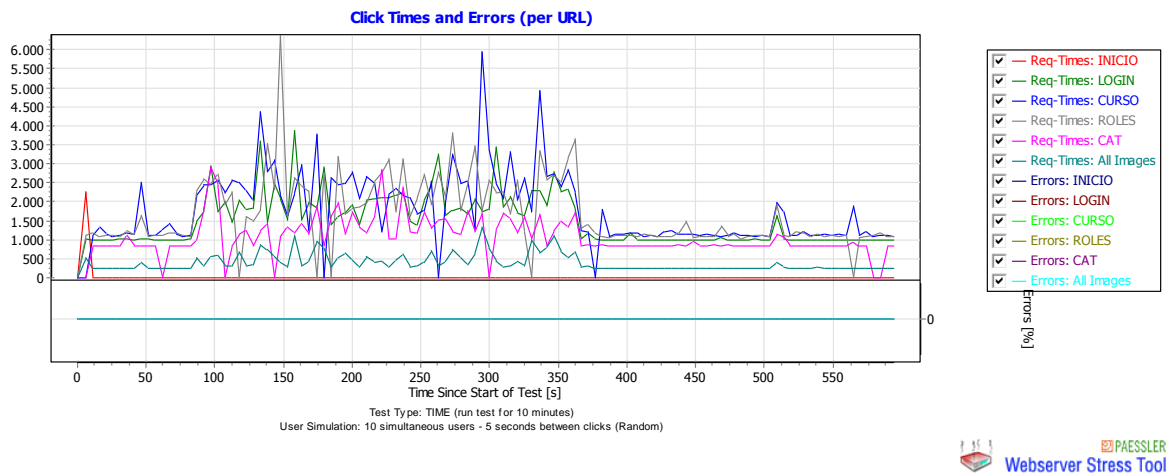


Figura 30. Prueba de errores por usuarios en cada url en prueba time

** Resultados por cada url **

URL#1 (INICIO): Average Click Time 2.265 ms, 10 Clicks, 0 Errors

URL#2 (LOGIN): Average Click Time 1.420 ms, 462 Clicks, 0 Errors

URL#3 (CURSO): Average Click Time 1.719 ms, 392 Clicks, 0 Errors

URL#4 (ROLES): Average Click Time 1.678 ms, 392 Clicks, 0 Errors

URL#5 (CAT): Average Click Time 1.105 ms, 413 Clicks, 0 Errors

Total Number of Clicks: 1.669 (0 Errors)

Average Click Time of all URLs: 1.478 ms

URL#1 (INICIO): Average Click Time 2.65 ms, 23 Clicks, 0 Errors

URL#2 (LOGIN): Average Click Time 1.2320 ms, 563 Clicks, 0 Errors

URL#3 (CURSO): Average Click Time 1.1319 ms, 125 Clicks, 0 Errors

URL#4 (ROLES): Average Click Time 1.1238 ms, 345 Clicks, 0 Errors

URL#5 (CAT): Average Click Time 1.545 ms, 354 Clicks, 0 Errors

Total Number of Clicks: 1.669 (0 Errors)

Average Click Time of all URLs: 1.534 ms

En el tipo de prueba Ramp, se realizó la prueba de 1 a 10 usuarios activos a una velocidad de 1 clic cada 5 segundos alternando las URL. Con esta prueba se comprobó la limitación de la aplicación a la carga de usuarios.

Los resultados fueron los siguientes:

Configuración para las pruebas.-

Tabla 16. Configuración para pruebas Ramp

Project and Scenario Comments, Operator	
Test Setup	
Test Type:	RAMP (run test for 10 minutes)
User Simulation:	ramp test with up to 10 simultaneous users - 5 seconds between clicks (Random)
Logging Period:	Log every 5 seconds
URLs	
URL Sequencing:	Users visit first 1 URL(s), then random URLs, then last 0
Browser Settings	
Browser Simulation:	User Agent: Mozilla/5.0 (compatible; Webserver Stress Tool 7; Windows)
Browser Simulation:	Simulate Maximum Data Rate: 120 kbit/s
Browser Simulation:	Enable Cookies
Recursive Browsing / HTML Parsing:	Download Images
Recursive Browsing / HTML Parsing:	Download EMBED, OBJECT, FLASH
Client System	
System	Windows NT V6.1 (Build 7601) Service Pack 1, CPU Proc. Lev. 686 (Rev. 9474) at 2133 MHz,
Memory	976 MB available RAM of 3072 MB total physical RAM, 2734 MB available pagefile, 20055 MB free disk space on C:
Test Software	
Webserver Stress Tool:	7.2.2.261 Trial Version

Resultados por usuarios

Tabla 17. Resultados por Usuarios en Prueba Ramp

User No.	Clicks	Hits	Errors	Avg. Click Time [ms]	Bytes	kbit/s
1	202	344	0	1.094	2.527.140	91,97
2	189	332	0	1.118	2.374.630	90,39
3	155	272	0	1.125	1.946.489	89,85
4	143	250	0	1.187	1.808.971	85,26
5	124	210	0	1.096	1.556.067	92,36
6	116	203	0	1.142	1.460.520	89,01
7	92	159	0	1.176	1.157.308	86,51
8	78	142	0	1.218	984.776	84,01
9	7 7	138	0	1.096	983.253	93,22
1 0	5 6	99	0	1.138	707.900	90,50

Resultados por url

URL#1 (INICIO): Average Click Time 2.65 ms, 23 Clicks, 0 Errors

URL#2 (LOGIN): Average Click Time 1.2320 ms, 563 Clicks, 0 Errors

URL#3 (CURSO): Average Click Time 1.1319 ms, 125 Clicks, 0 Errors

URL#4 (ROLES): Average Click Time 1.1238 ms, 345 Clicks, 0 Errors

URL#5 (CAT): Average Click Time 1.545 ms, 354 Clicks, 0 Errors

Total Number of Clicks: 1.669 (0 Errors)

Average Click Time of all URLs: 1.534 ms

Tabla 18. Resultado por url en Prueba Ramp

URL No.	Name	Clicks	Errors	Errors [%]	Time Spent [ms]	Avg. Click Time [ms]
1	INICIO	10	0	0,00	21.127	2.113
2	LOGIN	296	0	0,00	326.322	1.102
3	CURSO	301	0	0,00	391.331	1.300
4	ROLES	305	0	0,00	367.378	1.205
5	CAT	307	0	0,00	275.811	898

Gráficos de resultados

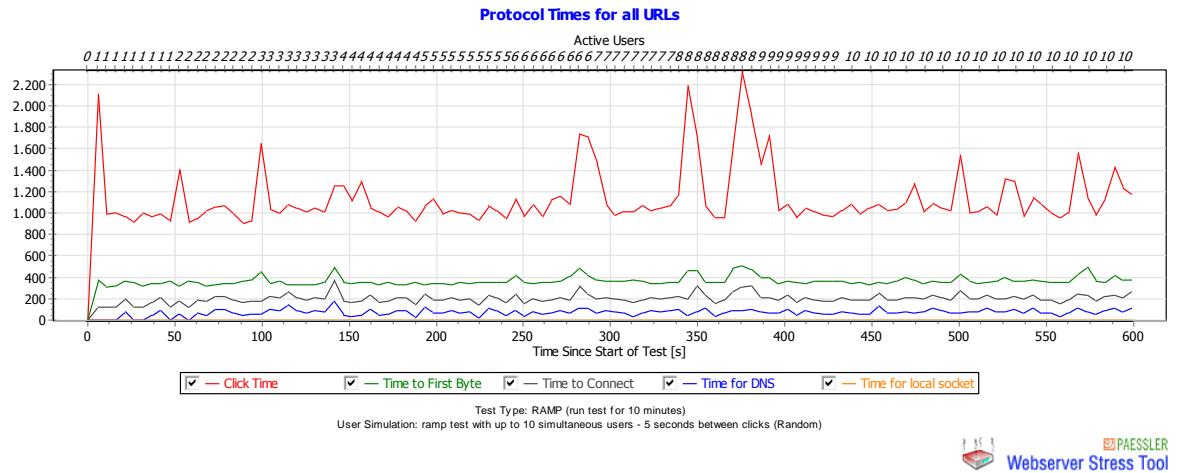


Figura 31. Tiempo de respuesta en prueba Ramp

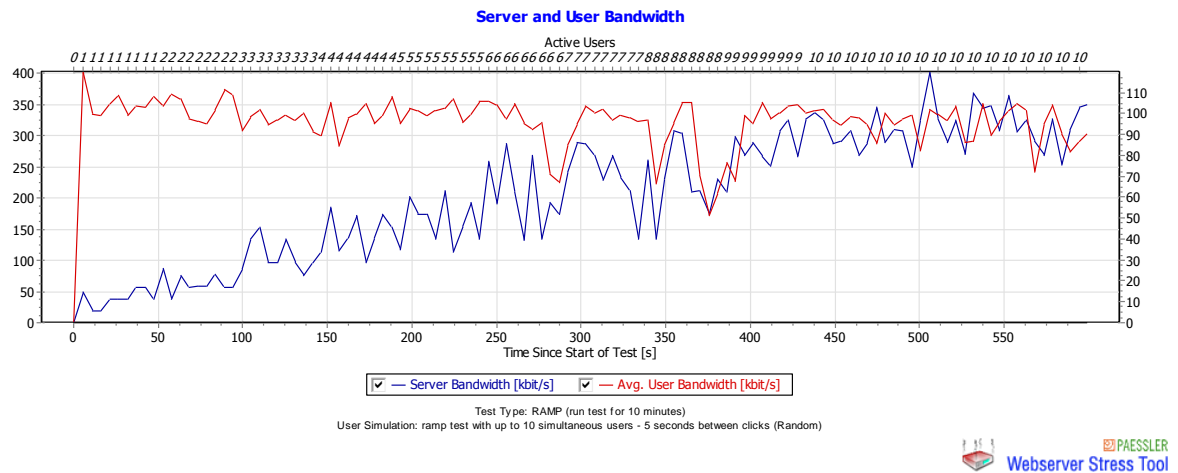


Figura 32. Uso de ancho de banda en prueba ramp

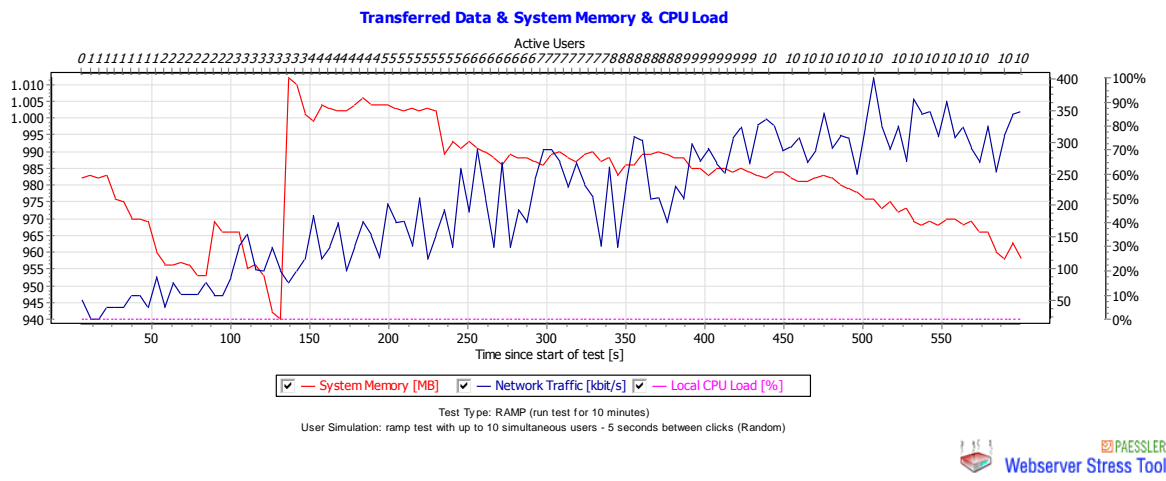


Figura 33. Uso del cliente en prueba ramp

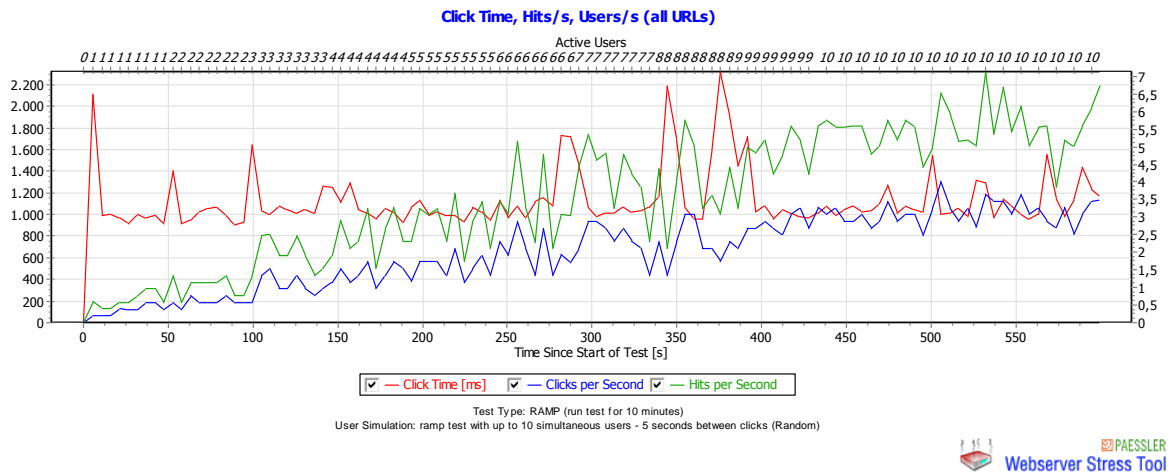


Figura 34. Promedio de espera en respuesta a requerimiento en prueba ramp

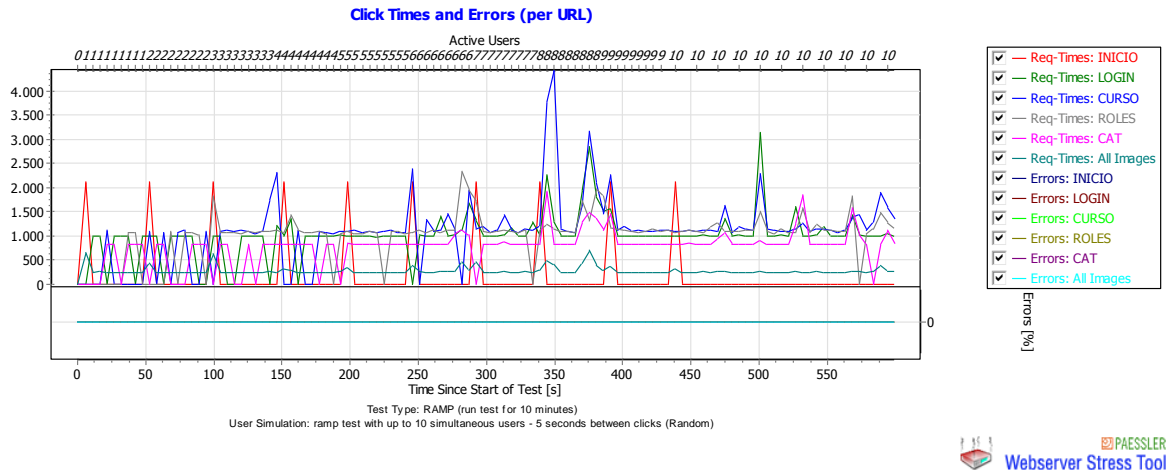


Figura 35. Prueba de errores por usuarios en cada url en prueba ramp

** Resultados por cada url **

URL#1 (INICIO): Average Click Time 2.113 ms, 10 Clicks, 0 Errors

URL#2 (LOGIN): Average Click Time 1.102 ms, 296 Clicks, 0 Errors

URL#3 (CURSO): Average Click Time 1.300 ms, 301 Clicks, 0 Errors

URL#4 (ROLES): Average Click Time 1.205 ms, 305 Clicks, 0 Errors

URL#5 (CAT): Average Click Time 898 ms, 307 Clicks, 0 Errors

Total Number of Clicks: 1.219 (0 Errors)

Average Click Time of all URLs: 1.134 ms

Los resultados dados una vez implantado el sistema, haber pasado las pruebas y puesto a producción son los siguientes:

Tabla 19. Resultados de implantación del sistema

Descripción	Valor	Observación
Instalación del Sistema	1	Instalación de MOODLE en un servidor dedicado.
Número de Usuarios Creados	4573	Creación de usuarios con carga masiva
Secciones Creadas	2	Creación de sección nocturna y diurna
Número de cursos creados	27	Creación de todos los cursos, de cada sección.
Instalación y parametrización de módulos en MOODLE	17	Módulos deseados para el trabajo óptimo del sistema, instalado.

4.2.4.2 DISCUSIONES

Como se observó en la Tabla 7 de Resultados de Implantación del Sistema, todo el alcance que tuvo la implementación del sistema fue superado, por ejemplo, el alcance está definido para los cuartos cursos de la sección nocturna, en camino a completar este objetivo y utilizando la carga masiva de usuarios se capacitó a dos cursos de estudiantes, los cuales se encargaron de generar los archivos de texto con la información necesaria para la creación de los usuarios de cada uno de los cursos, tomando esta iniciativa se logró la creación de todos los alumnos y profesores del colegio de las dos secciones, ya que la subida de estos archivos a moodle, no tomo más allá de una semana.

A pesar de que el alcance de la tesis se definió para un par de cursos de la sección nocturna la capacitación de los cursos de estudiantes para la creación de usuario y la capacitación del profesor Robert Montero como Usuario Administrador del sistema, permitió la implementación del sistema tanto para la sección Nocturna como la Diurna, esto se encuentra en producción y en uso de los usuarios dando un indicador favorable del cumplimiento del objetivo general de la tesis.

La instalación del sistema se la realizó en un servidor dedicado Linux con un servidor apache y la versión 1.9 de MOODLE, tiene un disco de 4 terabytes los cuales en base a cálculo son suficientes para dos años de carga de la institución.

Se realizó además la parametrización y creación de los módulos necesarios para el uso de los usuarios tanto profesores como de los alumnos, estos módulos fueron descargados de la pagina www.moodle.org los mismos pasaron por un proceso de descarga, e instalación el cual se encuentra en el punto 3.3.3 de la codificación, además de la realización de pruebas para verificar su correcto funcionamiento. Todos los módulos que se instalaron en MOODLE para el Colegio Intisana pasaron satisfactoriamente las pruebas y se encuentran en producción siendo usadas por los usuarios según su perfil.

CONCLUSIONES Y RECOMENDACIONES

5 CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

En el proceso de factibilidad técnica se realizó un estudio costo beneficio tecnológico, económico, operativo y legal, el cual demostró que la implantación del sistema es viable y que además se obtendría una gran ayuda para profesores y estudiantes.

Estos puntos pueden ser observados de mejor manera e individualmente en la planificación de la tesis, en donde se identificó que la implementación del sistema es factible y que su aporte a la institución será de gran importancia, para todos los usuarios así como un plus para la educación.

La instalación de la plataforma se realizó en un servidor dedicado donde se encuentra el host, y se configuró el sistema para que pueda ser utilizado acorde a las necesidades del instituto, para cumplir con esto se creó los usuarios, secciones, cursos, y los módulos necesarios para su uso acorde al perfil al cual cada usuario tiene acceso.

Todos los módulos, fueron descargados e instalados acorde a lo especificado, por el administrador del sistema del Colegio.

La capacitación fue realizada a dos grupos de estudiantes y una al administrador del módulo, los estudiantes no solo fueron capacitados en el uso del sistema si no también en la creación de archivos para la carga masiva de estudiantes, se capacitó al director de la sección nocturna el Sr. Robert Montero, quien fue de gran ayuda para el cumplimiento de los objetivos del sistema, siendo una persona muy entregada en obtener los conocimientos para realizar de mejor manera la administración del mismo.

5.2 RECOMENDACIONES

Como recomendaciones se propone los siguientes puntos:

- Mantener el sistema actualizado a la última versión estable para que este no tienda a convertirse en obsoleto.
- Verificar siempre que la capacidad del servidor se encuentre óptima para que no sufran los usuarios de lentitud en el uso del sistema o pérdida de información.
- Realizar una copia de seguridad por lo menos una cada semana y guardarle en otra computadora, y así, si el sistema o el servidor sufre un daño no perder la información y mantenerla siempre actualizada.

GLOSARIO

ERS Especificación de Requisitos de Software

FCI Facultad de Ciencias de la Ingeniería

AUTOMATIZACIÓN DE PROCESOS Es un sistema de fabricación diseñado con el fin de usar la capacidad de las máquinas para llevar a cabo determinadas tareas anteriormente efectuadas por seres humanos, y para controlar la secuencia de las operaciones sin intervención humana. El término automatización también se ha utilizado para describir sistemas no destinados a la fabricación en los que los dispositivos programados o automáticos pueden funcionar de forma independiente o semi-independiente del control humano.

BPM Business Process Management. Es el conjunto de servicios y herramientas que facilitan la administración de procesos de negocio, es decir análisis, definición, ejecución, monitoreo, y control de los procesos.

CU : CASOS DE USO Son parte del diseño, sino parte del análisis. De forma que al ser parte del análisis nos ayudan a describir qué es lo que el sistema debe hacer. Es lo que hace el sistema desde el punto de vista del usuario.

DOCUMENTO Es toda información fijada materialmente sobre un soporte, dicho soporte puede ir desde el clásico documento en papel a una fotografía, un documento creado mediante cualquier procesador de textos o una imagen digital.

DOCUMENTO INFORMÁTICO Son documentos producidos con la intervención explícita de equipo informático, y en consecuencia, altamente influenciados por cambios tecnológicos (por ejemplo, páginas HTML multimedia almacenadas en servidores distribuidos, entre otros).

DOCUMENTOS ELECTRÓNICOS Es un documento cuyo soporte material es algún tipo de dispositivo electrónico o magnético, y en el que el contenido está codificado mediante algún tipo de código digital, que puede ser leído, interpretado, o reproducido, mediante el auxilio de detectores de magnetización.

GESTORES DOCUMENTALES Programas de apoyo al proceso de gestión de la documentación que se maneja en la empresa. Existen soluciones

informáticas para: la gestión documental, administración de archivos digitales y para administración de bibliotecas de diversa índole.

PLAN DE TITULACIÓN Proceso sistemático, organizado y objetivo, cuyo propósito es responder a una pregunta o hipótesis y así aumentar el conocimiento y la información sobre algo desconocido.

PROCESO DE TITULACIÓN Es la última etapa de la formación profesional de un alumno a través de la cual demuestra que ha integrado los diferentes aspectos que conformaron su preparación profesional y la institución constata que el aspirante al título, es un profesionista en un campo de acción específico.

6 BIBLIOGRAFÍA

Becker, F. (05 de 02 de 2012). *Andes 2K - Gestión Documental*. Recuperado el 12 de 05 de 2012, de Andes 2K - Gestión Documental:
http://www.andes2000.cl/index.php?option=com_content&view=article&id=133&Itemid=180

Casmon, J. (14 de 06 de 2011). *Wikipedia - Gestión Documental*. Recuperado el 02 de 05 de 2012, de Wikipedia - Gestión Documental:
http://es.wikipedia.org/wiki/Gesti%C3%B3n_documental

IEEE Computer Society, I. (22 de 10 de 1998). *standards ieee*. Recuperado el 15 de 07 de 2012, de standards ieee:
<http://standards.ieee.org/findstds/standard/830-1998.html>

Pressman, R. (2005). *Ingeniería del Software, Un enfoque práctico*. En R. Pressman, *Ingeniería del Software* (Vol. VI, pág. 980). Mc Graw-Hill.

Sanchez, E. (21 de 02 de 2012). *Wikipedia - Proceso*. Recuperado el 18 de 06 de 2012, de Wikipedia - Proceso:
http://es.wikipedia.org/wiki/Proceso_%28inform%C3%A1tica%29

Sommerville, I. (2005). *Ingeniería del software*. Madrid: Pearson Educación, S.A.

Trejo, J. (19 de 03 de 2012). *Monografias.com - Base de datos*. Recuperado el 18 de 06 de 2012, de Monografias.com - Base de datos:
<http://www.monografias.com/trabajos11/basda/basda.shtml>

Trejejo Alonso, J. A. (2007). *Aprendiendo a Crear y Mantener Sitios Web*. Asturias: joomlaos.

PÁGINAS Y SITIOS WEB

Miriam Vasquez Cordoba, 2012

<http://www.slideshare.net/mirishvc/qu-es-una-plataforma-virtual>

(Enrique Martínez-Salanova, 2008)

<http://www.uhu.es/cine.educacion/didactica/0014procesoaprendizaje.htm>

(Fundación Wikimedia, Inc., 2012)

http://es.wikipedia.org/wiki/Educación_virtual

www.moodle.org

[Http://cursa.ihmc.us/rid=1176937289046_672363926_5091/mapa%20conceptual%20moodle.cmap](http://cursa.ihmc.us/rid=1176937289046_672363926_5091/mapa%20conceptual%20moodle.cmap)

<http://www.youtube.com/watch?v=36kpb4plvNA>

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

<http://www.e-historia.cl/e-historia-2/%C2%BFque-es-moodle/>

http://www.youtube.com/watch?v=fwlkTXoKh_s

http://www.youtube.com/watch?v=hTacy1_pSwU

ANEXOS

Anexo 2. Manual del Usuario

El manual de usuario debido a su extensión se encuentra anexo en un CD a la tesis.

El manual de usuario se encuentra los ingresos al sistema con pantallas que indican como navegar y realizar procesos básicos para la utilización del mismo.

Anexo 3. Esquema de la Base de Datos

mdl_assignment – Defines assignments

Field Name	Type	Len	Dec
-----	----	---	---
id			
course			
name			
description			
format			
resubmit			
type			
maxbytes			
timedue			
grade			
timeodified			

Total Bytes

mdl_assignment_submissions - Info about submitted assignments

Field Name	Type	Len	Dec
-----	----	---	---
id			
assignment			
userid			
timecreated			
timemodified			
numfiles			
grade			
comment			
teacher			
timemarked			
mailed			

mdl_backup_files - To store and recode ids to user and course files

Field Name	Type	Len	Dec
-----	----	---	---
backup_code			
file_type			
path			
old_id			
new_id			

mdl_backup_ids - To store and convert ids in backup/restore

Field Name	Type	Len	Dec
-----	----	---	---
backup_code			
table_name			
old_id			
new_id			
info			

mdl_chat - Each of these is a chat room

Field Name	Type	Len	Dec
------------	------	-----	-----

```

-----
id
course
name
intro
keepdays
studentlogs
chattime
schedule
timemodified
-----

```

mdl_chat_messages - Stores all the actual chat messages

```

=====
Field Name      Type      Len      Dec
-----
id
chatid
userid
system
message
timestamp

```

mdl_chat_users - Keeps track of which users are in which chatrooms

```

=====
Field Name      Type      Len      Dec
-----
id
chatid
userid
version
ip
firstping
lastping
lastmessageping
sid

```

mdl_choice - Available choices are stored here

```

=====
Field Name      Type      Len      Dec
-----
id
course
text
format
answer1
answer2
answer3
answer4
answer5
answer6
publish
timemodified

```

mdl_choice_answers

```

=====
Field Name      Type      Len      Dec
-----
id
choice
userid

```

answer
timemodified

mdl_config - Moodle configuration variables

```
=====
Field Name      Type      Len      Dec
-----
id
name
value
```

mdl_course - Course information

```
=====
Field Name      Type      Len      Dec
-----
id
category
sortorder
password
fullname
shortname
summary
format
showgrades
modinfo
newsitems
teacher
teachers
student
students
guest
startdate
numsections
showrecent
marker
visible
timecreated
timemodified
-----
```

mdl_course_category - Course categories

```
=====
Field Name      Type      Len      Dec
-----
id
name
description
parent
sortorder
coursecount
visible
timemodified
```

mdl_course_display - Stores info about how to display the course

```
=====
Field Name      Type      Len      Dec
-----
id
course
userid
```

display

mdl_course_modules - Stores info about active course modules

```
=====
Field Name      Type      Len      Dec
-----
id
course
module
instance
section
added
deleted
score
visible
```

mdl_course_sections - Stores info about individual course sections

```
=====
Field Name      Type      Len      Dec
-----
id
course
section
summary
sequence
visible
```

mdl_forum - Forums contain and structure discussion

```
=====
Field Name      Type      Len      Dec
-----
id
course
type
name
intro
open
assessed
assesstimefinish
scale
forcesubscribe
timemodified
```

mdl_forum_discussions - Forums are composed of discussions

```
=====
Field Name      Type      Len      Dec
-----
id
course
forum
name
firstpost
assessed
timemodified
```

mdl_forum_posts - All posts are stored in this table

```
=====
Field Name      Type      Len      Dec
-----
id
```

discussion
parent
userid
created
modified
mailed
subject
message
format
attachment
totalscore

mdl_forum_ratings - Contains user ratings for individual posts

```
=====
Field Name      Type      Len      Dec
-----
id
userid
post
time
```

mdl_forum_subscriptions -

```
=====
Field Name      Type      Len      Dec
-----
id
userid
forum
```

mdl_journal - Stores info about user journals

```
=====
Field Name      Type      Len      Dec
-----
id
course
name
intro
days
assessed
timemodified
```

mdl_journal_entries - All the journal entries of all people

```
=====
Field Name      Type      Len      Dec
-----
id
journal
userid
modified
text
format
rating
comment
teacher
timemarked
mailed
```

mdl_log - Every action is logged as far as possible

```
=====
Field Name      Type      Len      Dec
```

```

-----
id
time
userid
ip
course
module
action
url
info

```

mdl_log_display -

```

=====
Field Name      Type      Len      Dec
-----
module
action
mtable
field

```

mdl_modules - Stores info about each module loaded into moodle

```

=====
Field Name      Type      Len      Dec
-----
id
name
version
cron
lastcron
search
visible

```

mdl_quiz - Main information about each quiz

```

=====
Field Name      Type      Len      Dec
-----
id              Quiz id - unique number quiz attempted
course         Course the quiz is for
name          Name of quiz
intro         Message to user displayed under name
timeopen      Time quiz is first available to user
timeclose     Time quiz closes
attempts      Number of attempts allowed
attemptonlast Attempt builds on last: yes/no
feedback      Display feedback upon completion: yes/no
correctanswers Show correct answers in feedback: yes/no
grademethod   Grading method: i.e. highest, average
review        Allow users to review their answers: yes/no
shuffleanswers Shuffle answers so order is unique to each user: yes/no
questions
sumgrades
grade
timecreated   Time quiz was created
timemodified  Time quiz was last modified

```

mdl_quiz_answers - Answers with a fractional grade (0-1) and feedback

```

=====
Field Name      Type      Len      Dec
-----
id              Quiz id - unique number quiz attempted

```

question	Question id
answer	Correct answer
fraction	How right the answer is (0-1)
feedback	Feedback to display on grade report

mdl_quiz_attempts - Stores various attempts on a quiz

```
=====
Field Name      Type      Len      Dec
-----
id              INT       10
quiz           Quiz id - unique number quiz attempted
userid         INT       10      Quiz number
sumgrades      Number of questions answered correctly
timestart      INT       10      User id of student
timefinish     INT       10      Time quiz started
timemodified   INT       10      Time quiz finished
               Most recent time quiz grade modified
```

mdl_quiz_categories - Categories are for grouping questions

```
=====
Field Name      Type      Len      Dec
-----
id
course
name
info
publish
stamp
```

mdl_quiz_grades - Final quiz grade (may be best of several attempts)

```
=====
Field Name      Type      Len      Dec
-----
id              Unique number quiz attempted
quiz           Quiz number
userid         User id of student
grade          Final quiz grade displayed as a percentage
timemodified   Most recent time quiz grade modified
```

mdl_quiz_match - Defines fixed matching questions

```
=====
Field Name      Type      Len      Dec
-----
id
question
subquestions
```

mdl_quiz_match_sub - Defines subquestions that make up a matching question

```
=====
Field Name      Type      Len      Dec
-----
id              Quiz id number
question
questiontext   Text of question as it appears to student
answer        Corresponding text of answer as it appears to student
```

mdl_quiz_multianswers - Options for multianswer questions

```
=====
Field Name      Type      Len      Dec
-----
id              Quiz id number
question
answers
```


positionkey
answertype
norm

mdl_quiz_multichoice - Options for multiple choice questions

```
=====
Field Name      Type      Len      Dec
-----
id
question
layout
answers
single
```

mdl_quiz_numerical - Options for numerical questions

```
=====
Field Name      Type      Len      Dec
-----
id
question
answer
min
max
```

mdl_quiz_question_grades - The grade for a question in a quiz

```
=====
Field Name      Type      Len      Dec
-----
id
quiz
question
grade
```

mdl_quiz_questions - The quiz questions themselves

```
=====
Field Name      Type      Len      Dec
-----
id
category
name
questiontext
image
defaultgrade
qtype
stamp
version
```

mdl_quiz_randomsamatch - =====

```
=====
Field Name      Type      Len      Dec
-----
id
question
choose
```

mdl_quiz_responses -

```
=====
Field Name      Type      Len      Dec
-----
id
attempt
```

question
answer
grade

mdl_quiz_shortanswer - Options for short answer questions

```
=====
Field Name      Type      Len      Dec
-----
id
question
answers
usecase
```

mdl_quiz_truefalse - Options for true/false questions

```
=====
Field Name      Type      Len      Dec
-----
id
question
trueanswer
falseanswer
```

mdl_resource

```
=====
Field Name      Type      Len      Dec
-----
id
course
name
type
reference
summary
alltext
timemodified
```

mdl_scale - Defines grading scales

```
=====
Field Name      Type      Len      Dec
-----
id                                     Grade scale id
courseid         Course id number
userid           User id number
name
scale
description      Description of the rgade scale
timemodified     Most recent time grade scale
```

mdl_survey - Contains data for all surveys

```
=====
Field Name      Type      Len      Dec
-----
id
course
template
days
timecreated
timemodified
name
intro
questions
```

mdl_survey_analysis

```
=====
Field Name      Type      Len      Dec
-----
id
survey
userid
notes
```

mdl_survey_questions - Holds questions for each survey

```
=====
Field Name      Type      Len      Dec
-----
id
text
shorttext
multi
intro
type
options
```

mdl_user - Holds user information for for each person

```
=====
Field Name      Type      Len      Dec
-----
id              User id number
confirmed       Confirmed enrollment via e-mail: 1/0
deleted         Deleted user: 1/0
username        Username
password        Encrypted password
idnumber
firstname       User's first name
lastname        User's last name
email           User's e-mail address
icq             User's icq chat name
phone1          User's primary phone number
phone2          User's secondary phone number
institution
department
address         User's street address
city            User's city
country         User's country
lang            Language preferences chosen by
```

Anexo 4. Firmas de Capacitación a estudiantes