

**UNIVERSIDAD MAYOR DE SAN ANDRES
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMATICA**



Proyecto de Grado

**TEMA: SISTEMA CUSTOMER RELATIONSHIP MANAGEMENT (CRM)
LINGUA_UDABOL (Gestión de relaciones con clientes)**

DOCENTE (TUTOR) : LIC. EFRAIN SILVA
DECENTE (REVISOR) : LIC. GERMAN HUANCA
ALUMNO : ELIO RODOLFO CRUZ CHAMBI
CI. 4756030 LP

LA PAZ – BOLIVIA
2008

Dedico este trabajo a mi querida mamá Zenobia Chambi, por hacer realidad el deseo de ser profesional informático, por creer en mi, por darme su amor incondicional por los consejos prácticos y oportunos, y su invaluable paciencia y humildad que mostró en este trayecto.

Elio Rodolfo Cruz Chambi

Agradecimientos

A Jehová por dar sentido a la vida, de no ser por él no existiría creación alguna

Al licenciado Efraín Silva Sánchez, mi tutor, por la guía correcta en el desarrollo de este proyecto, con su experiencia y conocimiento en una tarea tan importante, en encaminar al profesionalismo.

Al licenciado German Huanca Ticona, mi revisor, por la paciencia, conocimiento, las sugerencias importantes que hicieron realidad este proyecto.

Al Centro de enseñanza de Idioma Ingles, Universidad Aquino Bolivia, por darme la oportunidad de implementar y desarrollar el proyecto CRM. Gracias por la confianza depositada en mi persona.

A mi tía Petrona Chambi por la confianza, su ayuda incondicional en todo sentido, también agradecer a mi prima cecilia.

A mis buenos y queridos amigos Ronald, Cristhian, Reynaldo, yeyo.

Y a Cinthia, una persona muy especial que robo mi corazón y ocupa un lugar importante en mi vida, dio sentido y dirección a mi vida, siendo la columna en mi estudio de la Universidad.

A la Facultad de Ciencias Puras y Naturales y la Carrera de informática por abrirme las puertas a un éxito profesional.

Elio Rodolfo Cruz Chambi

Referencia sobre el nombre del proyecto

El título del proyecto de grado SISTEMA CUSTOMER RELATIONSHIP MANAGEMENT (CRM) LINGUA_UDABOL significa (Gestión de relaciones con clientes).

El modelo de trabajo de todo el centro de enseñanza del lenguaje extranjera, es un modelo empresarial que tiene como objetivo el seguimiento minucioso a los clientes, en este caso a los estudiantes como a los interesados en tomar los cursos que la institución Lengua_Udabol presta servicios. Por lo tanto el nombre se adopto por el método ya utilizado en la institución.

En conformidad con lo mencionado para los administradores y responsables del funcionamiento de toda la institución, el sistema será el apoyo a este modelo de trabajo, con aspectos necesarios del cliente.

Los detalles en el sistema de los estudiantes serán necesarios para un buen control e interés en los clientes, el sistema hará el control de aspectos como los pagos, moras inscripción a las materias horarios, Etc. Será un modelo de aproximación académico. El sistema realiza lo necesario para no perder el control de datos de los estudiantes inscritos en la institución.

Por ser un modelo empresarial donde las mismas se encuentran en el Internet para ser conocidos, también el sistema estará en la Web como información a los clientes.

Resumen

El proyecto pretende plasmar datos importante y necesarias de los clientes desde la primera entrevista en la institución, centro de enseñanza del lenguaje extranjera Lingua_Udabol, para llevar un control de manejo de información sobre el cliente, el cual ayudara a coadyuvar en decisiones elementales, siendo este software una herramienta muy necesario y oportuna. Será el complemento al modelo utilizado “Gestión de Relación con Cliente” en esta institución, La repercusión del sistema trascenderá en principio en La Paz posteriormente a nivel Nacional.

Se empleara la metodología denominado Programación Extrema (XP), el cual es una herramienta velos para el desarrollo del sistema.

Para hacer realidad este sistema se utilizo las herramientas como el Apache como servidor de paginas, PHP como intermediador con el gestor de la Base de Datos MySql, también JavaScrip, HTML para la estética y estilo de las paginas del sistema llamado SISTEMA CUSTOMER RELATIONSHIP MANAGEMENT (CRM) LINGUA_UDABOL.

Para las metricas de calidad la norma **ISO 9126** es un estándar internacional para la evaluación del Software.

INDICE GENERAL

Dedicatoria.....	i
Agradecimiento.....	ii
Resumen.....	iii
Referencia sobre el nombre del proyecto.....	iv
Capitulo 1 Introduccion.....	1
Capitulo 2 Marco Teorico.....	12
Capitulo3 Marco Aplicativo.....	35
Capitulo4 Conclusiones y Recomendaciones.....	62
Referencias bibliograficas.....	
Referencia Bibliográfica	
Glosario	
Anexos	



CAPITULO 1

1. INTRODUCCION

1.2. Antecedentes.....	2
1.2.1 De la institución.	2
1.2.2 Información referencial.....	2
1.3. Identificación del problema.....	6
1.4 Objetivos.....	7
1.4.1. Objetivo General.....	7
1.4.2. Objetivos Específicos.....	7
1.5. Justificación.....	8
1.5.1 Justificación económica.....	8
1.5.2 Justificación social.....	8
1.5.3 Justificación técnica.....	8
1.6. Métodos y técnicas.....	8
1.6.1 Programación Extrema.....	8
1.7. alcances y aportes.....	11

CAPITULO 2

MARCO TEORICO

2.1 Sistemas de gestión.....	12
2.1.1 Fases del Control de Gestión.....	12
2.2. C.R.M (gestión de relación con cliente).....	15

2.2.1 La importancia de una estrategia CRM en una Empresa.....	15
2.2.2 Software de CRM.....	16
2.2.3 Aplicaciones de CRM.....	18
2.2.4 Cómo aprovechar las ventajas de la <u>comunicación</u> Electrónica.....	19
2.3 Metodología XP (programación extrema).....	19
2.3.1 Fundamentos de Programación Extrema.....	20
2.3.2 Los pasos a seguir en un proyecto	21
2.3.3 Características.....	22
2.4 Usos y aplicaciones de XP.....	30
2.5 Ventajas y desventajas de extreme programming.....	30
2.6 Tecnología	31
2.6.1 Definiciones.....	31
2.7 Herramientas.....	32

CAPITULO 3

3 MARCO APLICATIVO	35
3.1 Programación Extrema(xp).....	35
3.2 Fase de iteración.....	44
3.3 Diagrama Navegacional.....	46

3.4 Esquema de análisis para la gestión de claves.....	48
3.5 Calidad de Software	55
3.6 Análisis de beneficio.....	60

CAPITULO 4

Conclusiones y recomendaciones

4.1 Conclusiones.....	62
4.2 Recomendaciones.....	63

Referencia Bibliográfica

Glosario

Anexos



INDICE DE FIGURAS

Figura N° 1 Inscripción de los estudiantes (de lengua idioma).....	5
Figura N° 2 Registro de pagos de los estudiantes.....	5
Figura N° 3 Estructura de las tablas de la base de datos de Lengua_Udabol.....	43
Figura N° 4 CRC Correspondiente a la Historia N° 1.....	45
Figura N° 5 CRC Corresponde a usuario N° 2.....	45
Figura N° 6 Diagrama navegacional del sistema.....	47
Figura N° 7 Análisis para la gestión de clave	48
Figura N° 8 Test.....	51



INDICE DE TABLAS

Tabla N° 1 Participación en el mercado.....	19
Tabla N° 2 Equipo de Proyecto.....	35
Tabla N° 3 Iteración del sistema.....	44
Tabla N° 4 Glosario de términos.....	46
Tabla N° 5 Proceso de migración de datos.....	49
Tabla N° 6 Declararon de métodos de evaluación.....	50
Tabla N° 7 Parámetro de medición del sistema.....	52
Tabla N° 8 Factores de intervención para medición.....	53
Tabla N° 9 Árbol de requerimiento de calidad.....	56
Tabla N° 10 Resultado de las Preferencias de Funcionabilidad.....	57
Tabla N° 11 Resultado de las Preferencias de Usabilidad.....	58
Tabla N° 12 Resultado de las Preferencias de Eficiencia.....	59
Tabla N° 13 Resultado de las Preferencias de Confiabilidad.....	59
Tabla N° 14 Resultados de las preferencia global.....	60

CAPITULO 1

SISTEMA CUSTOMER RELATIONSHIP MANAGEMENT (CRM)

LINGUA_UDABOL (Gestión de relaciones con clientes)

1. INTRODUCCION

El transcurrir del tiempo ha enseñado al ser humano la necesidad de guardar datos de la concurrencia realizado a instituciones y centros de enseñanza.

Trasciende más cuando se trata de controlar la cantidad de dicha concurrencia a centros de enseñanza del idioma inglés. Por tanto existen muchas formas de controlar la actividad de recepción que desempeña el responsable del funcionamiento de la institución, para elaborar el control de su labor crearon formularios, planillas y otros medios manuales; por la importancia de los datos a ser guardados los formularios como documentos es posible guardarlos, pero al transcurrir del tiempo el volumen de estos documentos es abundante y si se trata de buscar información de estos papeles es moroso. La manera más práctica es la utilización de la tecnología, reduce el tiempo y espacio.

En los centros de enseñanza de otros idiomas en aspecto académico y recepción de personas es muy necesario, la utilización de un sistema hará posible la facilidad de encontrar información y datos importantes del estudiante (cliente) y también de los administradores, por medio de una base de datos hará posible gestionar procesos que involucre a alumnos recepcionados y después informes de cada uno de ellos.

En la actualidad cuenta con un sistema hecho en Excel, el cual imposibilita buscar respuestas a preguntas frecuentes de estudiantes como de materias aprobadas de los módulos, las diversas formas de atención al cliente los módulos por cursar y los módulos pagados.

Por esta razón la institución opta por un sistema automatizado, el cual permitirá realizar tareas de inscripción, horarios, módulos a cursar, esto permitirá tener mayor control de seguimiento administrativo con resultados satisfactorios para todo el centro de educación.

2. ANTECEDENTES

Para la elaboración del proyecto se recavo información de la misma institución y del Internet, donde nos dará a saber la perspectiva del funcionamiento del centro de educación del idioma ingles, para una mejor atención al cliente.

2.1 De la institución.

Centro de enseñanza del idioma ingles de UDABOL cuya sigla de método de trabajo CRM se basa en la enseñanza de lenguaje extranjera,

La metodología utilizada por LINGUA IDIOMAS esta basada en el **NA-SLA+L** (Natural Approach to Second Language Acquisition and Lexical Development).

Este método, reúne los principios y estrategias de los dos principales enfoques en la enseñanza moderna de idiomas:

- El Enfoque Natural (Krashen y Terrell, 1977)
- El Enfoque Léxico (Lewis, 1993)

El primero representa el método con mayor influencia en enseñanza de idiomas en Estados Unidos y el mundo entero.

Su visión del idioma es enteramente comunicativa y al igual que el enfoque léxico pone un énfasis particular en el principio que el lenguaje es un conjunto de mensajes que comprenden y reproducen.

El programa está diseñado en base a cuatro niveles los cuales conducen a una Certificación de Técnico Superior en el Idioma Inglés. Al concluir con el tercer nivel, al estudiante le es otorgado un Certificado de Técnico Medio.

NIVEL 1: INGLÉS BÁSICO

FIRST CERTIFICATE IN ENGLISH - 4 Módulos

Al concluir el nivel, el personal tendrá conocimientos suficientes para poder comunicarse en situaciones básicas, otorgar y recibir información no compleja verbal y/o escrita.

NIVEL 2: INGLÉS INTERMEDIO

CERTIFICATE OF INTERMEDIATE ENGLISH - 4 módulos

Al concluir el nivel el personal podrá comprender el idioma inglés hablado, leer textos de interés general, escribir con mediana complejidad y comunicarse oralmente de forma efectiva.

NIVEL 3: INGLÉS AVANZADO

TÉCNICO MEDIO EN INGLÉS - 4 módulos

Al concluir el Programa, el personal habrá desarrollado un conocimiento avanzado del idioma inglés, esto significa que podrá comunicarse eficientemente de forma oral y escrita en cualquier circunstancia.

NIVEL 4: ESPECIALIZACIONES:

TÉCNICO SUPERIOR EN INGLÉS – 4 módulos

El más grande modelo de enseñanza de un idioma es el proceso de aprendizaje de la lengua materna y de la experiencia de su estudio se puede concluir lo siguiente:

1. El desarrollo del lenguaje es un resultado de la necesidad de comunicación y de expresión.
2. El aprendizaje de un idioma es un proceso comunicativo en el que la interacción es indispensable. No se puede desarrollar un nuevo idioma en aislamiento.
3. La estructura temática debe ser funcional, por lo que el uso de situaciones reales es indispensable. En el proceso de enseñanza de un idioma debe contemplarse la gramática como complemento, no así como base del aprendizaje.
4. La motivación y el involucramiento emocional con el idioma meta son indispensables para la comprensión y la producción del mismo.

Es importante mencionar que el método es utilizado por toda la academia de forma estricta mediante el uso de un Plan de Lección estructurado en base a principios pedagógicos uniformes y actividades comunicativas donde los estudiantes interactúan sin ninguna dificultad.

DURACIÓN DE LOS CURSOS:

- a) Cursos Regulares: 2 años de estudio (3 niveles, Técnico Medio)
- b) Cursos Intensivos: 1 año de estudio (3 niveles, Técnico Medio)

2.2 Información referencial

El sistema como herramienta de trabajo utilizado desde el comienzo de la institución es la que daremos a conocer a continuación.

Figura N° 1 Inscripción de los estudiantes (de lengua idioma)

	A	B	C	D	E	F	G	H
1								
2		NOMINDA DE ALUMNOS DE LINGUA						
3								
4	1	Achu	Cocarico	Orlando				
5	1	Aguilar	Contreras	Jose Pablo				
6	1	Aguilar	Espinoza	Paola Teresa				
7	1	Aguifera	Fernandez	Pool Diego				
8	1	Aguirre	Vargas	Carla Susana				
9	1	Aima	Vertiz Blanco	Claudia Evelin				
10	1	Alanes	Morales	Maria Alejandra				
11	1	Alarcon	Pomier	Roger Igor				
12	1	Alarcon	Saravia	Marcelo Alberto				
13	1	Alba	Moscoso	Betania angela				
14	1	Albarracin	Peralta	Boris Jorge				
15	1	Ali	Marnani	Maria Eugenia				
16	1	Allaga	Del Carpio	Javier				
17	1	Allaga	Del Carpio	Adriana				
18	1	Aller	Salcedo	Piedad				
19	1	Altamirano	Cardenas	Elias				
20	1	Alvarez		Iver				
21	1	Alvarez	Alconini	Fabricio				
22	1	Amonzabel	mariscal	Carla				
23	1	Andrade	Calvimontes	Sergio Yamil				

Figura N° 2 Registro de pagos de los estudiantes

	A	B	C	D	E	F
182	20					
183	18					
184	8	REGISTRO DE PAGOS MENSUALES				
185		Teacher's: Luis Vasquez		STARTER II TURNO B (10 de Marzo) de 09:00 a 12:15		
186		DATOS PERSONALES		DATOS DOMICILIO		
187	No.	APELLIDO PATERNO	APELLIDO MATERNO	NOMBRES	DIRECCION	
188	1	Melena	Nosa	Flaviana	C. Capitan Castrillo No. 226 Sopocachi	
189	2	Ortiz	Dorado	Zdenka Pamela	Calle san Ramon No. 100B Achumani	
190	3	Barral	Espejo	Walter julio	Calle Elias Sagarnaga No. 1028	
191	4	Soliz	gutierrez	Yoselln	C. Carreras No. 845 Sur	
192	5	Rios		Karen		
193	6	Auza	Auza	Lourdes Daniela	Av. Busch No. 1173 Miraflores	
194	7	Ramirez	Benavides	Edgar Reynaldo	Av. La Paz No. 17 el Alto	
195	8	Ouispe		Blanca		
196	9	Perez	Chavez	Cynthia Rosangela	Av. Alfonso Ugarte No. 54 16 de Julio	
197	10	Herrera	Clavijo	Jenny Susan	Calle 28 No. 13 Achumani	
198	11	Rodriguez	Montero	Georgina Isabel		
199	12	Morales	Parra	Conzuelo Concepc	Av. Costanera No. 20 V. San Antonio Bajo	
200	13			Wendy		
201	14	Alvarez	Zalles	Alejandra Mellie	Calle Resequin No. 2012 Sopocachi	
202	15	Zabala	Tatiana	Erika Tatiana	Calle 6 No. 715 Obrajes Ricon Alto	

En cuanto a los proyectos realizados con relación al sistema de información de seguimiento al estudiante, no existe mucha información, sin embargo se puede citar trabajos realizados en Excel como vimos en las pantallas anteriores con algunos datos que puede ser importante para la elaboración del sistema en el proyecto de grado.

3. IDENTIFICACION DEL PROBLEMA

El problema que presenta la institución es la falta de atención minuciosa al cliente. Por ser parte de la universidad Udabol, el centro de idioma ingles utiliza el mismo sistema académico para sus notas de las materias tanto aprobadas como reprobadas y como el record académico.

Para dar prioridad y atención al cliente, falta un sistemas de control para el manejo de datos de cada estudiante o cliente, cubrir las necesidades para el buen funcionamiento de la institución y así no perder la misma.

Los problemas mas frecuentes identificados de esta institución son las siguientes:

- No tener acceso inmediato a la información de parte de las autoridades administrativas sobre el manejo de datos de los clientes.
- El sistema de seguimiento al cliente que se utiliza todavía no cuenta con la información suficiente para los estudiantes y administrativas.
- La inseguridad de manipuleo de los datos en el sistema utilizado.
- Consultas de las notas por modulo de los estudiantes.
- Comunicación directa con Caja.
- Las materias pagadas por los diferentes módulos.
- Abandono de módulos.
- Control de pagos de moras.
- Si se dio seguimiento al cliente desde el inicio.
- Consulta frecuente de los horarios y aulas.

Para identificar los problemas mas frecuentes las causas y efectos se utilizo el cuadro del análisis de problemas ver (anexo # 1)

4 OBJETIVOS

4.1. Objetivo General

Automatizar los procesos administrativos y manejo de clientes del Centro Lingua_Udabol de la Universidad Aquino Bolivia.

4.2. Objetivos Específicos

- Realizar el análisis de requerimientos ventajas y desventajas.
- Implementar el aspecto administrativo de la pre-inscripción a los cursos de lengua extranjera.
- Implementación de la base de datos para guardar la información más importante de los estudiantes.
- Realizar las medidas máximas de seguridad de los datos de los clientes del centro de idiomas.
- Generar módulos de consulta de materias pagadas y por pagar.
- Generar reportes de los horarios asignados por módulos y aulas.
- Implementación un modulo del control de moras de los estudiantes.
- Implementar el control de pagos realizados por módulos.
- Realizar un modulo de asignación de docentes y aulas.
- Realizar seguimiento minucioso al cliente tanto alumno activo como inactivo.

5. JUSTIFICACION

5.1 JUSTIFICACION ECONOMICA

El sistema ayudara a reducir el tiempo para recopilar información de las operaciones administrativas, con relación a informes y reportes de los estudiantes; otro factor importante es la reducción del material de escritorio, esto trascenderá significativamente en reducir el costo de trabajo con relación a horas/hombre del personal.

5.2 JUSTIFICACION SOCIAL

El sistema será de gran beneficio para los interventores directos como los administradores y los de recepcionista de clientes, que son los responsables de proporcionar diferente tipo de información a los estudiantes. También el estudiante tendrá intervención directa con el sistema.

5.3 JUSTIFICACION TECNICA

El sistema cuenta con la tecnología necesaria para el desarrollo del proyecto con una computadora que pueda responder a las actividades necesarias de toda la gestión académica, en cuanto al Software que se utilizara para la implementación del sistema será PHP y MySQL siendo un software libre y sin costo adicional.

6. MÉTODOS Y TECNICAS

6.1 Programación Extrema

Es una *metodología de programación* que, en muchos casos, hace las cosas bastante más sencillas.

¿En qué se diferencia entonces la XP de otras metodologías?

En la XP, la importancia de los test (para encontrar errores) y la experiencia del cliente (feedback) adquieren una mayor importancia, haciendo que se trabaje en ciclos de menor tiempo. Tranquilos que lo explicaremos todo.

Importancia de toda la teoría

Sólo hay que ver que se inventaron los atributos públicos para las variables en lenguajes orientados a objetos, cuando "en teoría" todas deberían de ser privadas y controladas mediante setters y getters. Eso no significa que no piensen antes de empezar un proyecto. Es bueno (y necesario) hacer un mínimo esquema de cómo deben de ser las cosas y funcionar el sistema, pero no pasar horas y horas de organización (todo dependerá del tamaño del proyecto). Recuerda que poca organización es mala, pero en exceso también.

Los tests antes que el programa

La XP llevada al extremo implica que se escriban los test (debug tests) antes que la propia aplicación. Esto tampoco es estrictamente necesario, pero sin embargo hay "Tests Units Frameworks" que son una serie de tests pre-fabricados para aplicar directamente a tu aplicación. ¿Por qué? ¿Por qué son tan importantes? Cuando se programa de la forma old-school, uno deja para el final el testear su aplicación y encontrar errores. Esto significa que se le da una menor importancia porque "en teoría" el programa no debería de fallar, aunque la realidad sea otra. Además suele ocurrir que al encontrar un fallo se aplica un parche que sólo ensucia el código y que ni siquiera se integra bien con el código.

Sed sinceros, ¿cuántos de vosotros ha creado variables y métodos especiales para solucionar un fallo, cuando podría haber utilizado parte de los recursos a existentes? ¿Y después nos olvidamos de esas variables, sin borrarlas ni nada?

Testeando nuestra aplicación desde el primer momento, nos aseguramos de que el código escrito hasta entonces es correcto y no estropea lo anterior. La programación orientada a objetos facilita esta tarea, ya que los objetos tienen una "interfaz" y deben responder ante ciertos "estímulos". Así por ejemplo, en un caso práctico, podríamos escribir los métodos vacíos de una clase, dentro de cada método escribir solamente el código necesario para que imprima por pantalla un mensaje que diga que se llegó a tal método o a tal otro. Y después, poco a poco, ir escribiendo el resto del código.

Consulta con el cliente

El feedback es importante, y la XP lo potencia al máximo. Si el programador idea una forma de ir desde A hasta C pasando por B, el cliente o el usuario encontrará la forma de ir directamente de A a C generando errores por el camino. A cada poco se suelta una beta para testeo, con la información resultante de los testeadores se hace una "pequeña" release. Pequeños cambios implican pequeños errores que se solucionan en menor tiempo. Esto va acorde con el de hacer pequeños módulos en el programa, y se van probando los módulos uno a uno según su desarrollo.

Conclusiones

La XP está ahí afuera. Es un recurso útil y al igual que muchos de los patrones de programación (MVC, Singleton...) posiblemente estés usando ya parte de la XP en tus proyectos a diario.

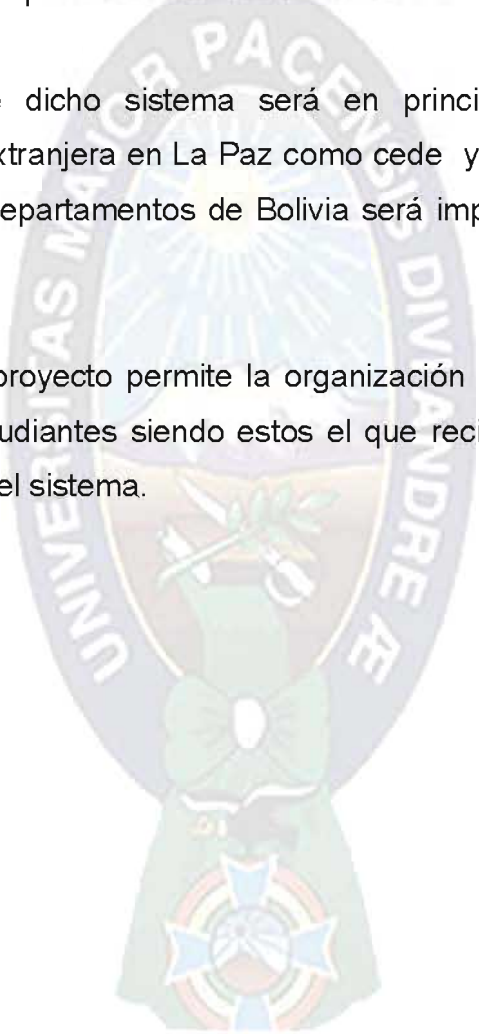
Pero nunca viene mal conocer algunos consejos extra para hacernos más ágiles en el desarrollo de software.

7. ALCANCES Y APORTES

El proyecto de grado tiene como objetivo realizar las especificaciones de requerimientos, Software para el centro de estudio del idioma inglés.

La implementación de dicho sistema será en principio para el centro de enseñanza del idioma extranjera en La Paz como sede y posteriormente si existe la posibilidad en otros departamentos de Bolivia será implementada de la misma manera.

El aporte del presente proyecto permite la organización minuciosa de las tareas administrativas y de estudiantes siendo estos el que recibirá el mayor beneficio con la implementación del sistema.



CAPITULO 2

2. MARCO TEORICO

En este capitulo se especifica los fundamentos teóricos principales para el desarrollo del proyecto a partir de recopilación de informes hasta la misma implementación del producto.

2.1 SISTEMAS DE GESTIÓN

Sistemas de Gestión Las empresas, independientemente de su tamaño, enfrentan demandas respecto a rentabilidad, calidad, tecnología y desarrollo sostenible. Un sistema de gestión eficiente, diseñado a la medida de sus procesos comerciales, puede ayudar a enfrentar los desafíos del cambiante mercado global de hoy.

Para convertir las presiones de la competencia en ventajas comparativas, las empresas deben aumentar el rendimiento operativo en forma sistemática. Un sistema de gestión puede ayudar a centrar, organizar y sistematizar los procesos para la gestión y mejora.

2.1.1 Fases del Control de Gestión

Resulta importante, para poder realizar un diagnóstico de la situación del Sistema de Control de Gestión de una organización, destacar un grupo de aspectos que definen el nivel o grado de complejidad y conformación de los sistemas de control de gestión atendiendo a un grupo de características:

Calidad y confiabilidad en la contabilidad, cultura organizativa.

Complejidad de la actividad, posibilidad de formalización del proceso, características del entorno, tipo de estructura.

Especialización de la actividad y de los centros de responsabilidad, un mayor peso en factores de carácter estratégico como planes, presupuestos, proyectos, etc.

Según Jordan (1997), "el Control de Gestión pasa por distintas fases en su aplicación, en dependencia de las características de cada organización y los objetivos propuestos por éstas."

Fase 0: Situación inicial.

Contabilidad General, producida anualmente con orientación fiscal. Retrasos de seis meses desde el cierre del ejercicio. Documentos de gestión puntuales. No existe la contabilidad de gestión.

Fase 1: Acciones simultanea.

Se mejora en algo lo existente. Contabilidad operativa.

Fase 2: Acciones simultaneadas.

Creación y puesta en marcha de una Contabilidad Analítica, lo que supone la identificación y formalización de Centros de Responsabilidad a la vez que la consideración de los Factores Críticos del Éxito del negocio.

Fase 3: Puesta en marcha de previsiones a corto plazo.

Presupuestos anuales por meses.
Estados Financieros anuales provisionales.

Fase 4: Acciones simultaneadas.

Elaboración de un plan estratégico "deslizante", esencialmente financiero, por centros de responsabilidad.

Integración del presupuesto a un año en el plan estratégico, como primer año del plan.

Fase 5: Extensión del proceso presupuestario a dos o tres años.

El plan operativo se alarga. Existen planes de acción con su traducción a términos financieros.

Fase 6: Concordancia en el tiempo de los elementos siguientes:

Plan estratégico.

Plan operativo.

Presupuesto.

Seguimiento de resultados presupuestario como proceso de aprendizaje.
Integración del seguimiento de resultados al proceso provisional. Cierta flexibilidad en las fechas de elaboración de planes.

Fase 7: Integración formal.

Integración formal de los resultados presentes en la definición de los planes de acción.

Consolidación de los Centros de Responsabilidad.

Fase 8: Delegación de responsabilidades de gestión.

Utilización de un sistema de seguimiento provisional integrado bajo el aspecto de delegación de responsabilidades de gestión. Cambio de orientación en los criterios de resultados.

Es necesario destacar que no todas las organizaciones deben estar en la fase más alta e incluso no tienen necesidad de encontrarse en las de mayor nivel. Todo depende de las características de la organización en específico.

2.2. CRM (GESTIÓN DE RELACIÓN CON CLIENTE)

CRM (Customer Relationship Management), en su traducción literal, se entiende como la Gestión sobre la Relación con los Consumidores, pero es tan genérico como toda frase en inglés traducida al español. Pero para su mejor comprensión básicamente se refiere a una estrategia de negocios centrada en el cliente.

"CRM estamos frente a un modelo de negocios cuya estrategia esta destinada a lograr identificar y administrar las relaciones en aquellas cuentas más valiosas para una empresa, trabajando diferentemente en cada una de ellas de forma tal de poder mejorar la efectividad sobre los clientes". En resumen ser más efectivos al momento de interactuar con los clientes.

Bajo este concepto, sería bueno profundizar, ya que estas tres palabras incluyen mucho más. El CRM como lo define consiste en ciertos componentes:

Funcionalidad de las ventas y su administración

- El telemarketing
- El manejo del tiempo
- El servicio y soporte al cliente
- El manejo de la información para ejecutivos
- La integración del ERP (Enterprise Resource Planning)
- La excelente sincronización de los datos

Sin embargo la palabra lealtad, sintetiza prácticamente su significado, ya que CRM se dedica a adquirir y mantener la lealtad del cliente.

2.2.1 La importancia de una estrategia CRM en una empresa

La competencia no permite que se descuide al protagonista de la película, el cliente.

Una implementación efectiva de CRM permite mejorar las relaciones con los clientes, conociéndolos mejor y permitiendo disminuir los costos en la consecución de nuevos prospectos y aumentar la fidelidad de los ya existentes, lo cual, en ambos casos, significa mayores ingresos y más rentabilidad para el negocio. Además, también se obtienen beneficios relacionados con mejores resultados en el lanzamiento de nuevos productos y en el desarrollo de marketing más efectivo. CRM, Customer Relationship Management, se refiere a la administración de todas las interacciones que pueden tener un negocio y sus clientes. Se enfoca en la optimización del ciclo de vida del cliente en su totalidad. Además, CRM es un término de la industria de la información que reúne metodologías, software y las capacidades de la Internet para administrar de una manera eficiente y rentable las relaciones de un negocio con sus clientes.

2.2.2 Software de CRM.

Desde el primer contacto hasta el servicio posterior a la venta, las soluciones de CRM en Web permiten darle seguimiento a las actividades de los clientes, mejorar la efectividad de ventas, proporcionar un mejor servicio al cliente y crear relaciones rentables con los clientes.

Existe una buena razón de por qué Customer Relationship Management (CRM) está en boca de todos. En un mundo donde los productos son cada vez más iguales, la estrategia corporativa que triunfe será la que comprenda que la atención que proporciona a sus clientes es el factor decisivo para atraer y mantener a los clientes.

CRM puede ayudar a una organización a transformar la forma en que mercadea, vende y ayuda a sus clientes. Con un enfoque de CRM, una organización utilizará cada oportunidad que tenga para encantar a sus clientes, fomentando con esto su lealtad y construyendo relaciones a largo plazo mutuamente satisfactorias.

Algunos ejemplos de los softwares más utilizados son:

PeopleSoft: Esta solución transforma datos complejos en información útil, lo que facilita tomar decisiones en tiempo real que incrementen la rentabilidad del cliente. Permite además que ejecutivos de mercadotecnia, ventas y soporte sean más estratégicos en la manera como manejan la interacción con sus clientes. Por ella identifican a los clientes que compran o que no están interesados en hacerlo. Pueden además: incrementar los ingresos en el punto de interacción, reducir la carga laboral y aumentar el enfoque en los proyectos más estratégicos; así como enfocarse de manera efectiva a las inversiones y a los mensajes mas importantes y reducir la longitud de los ciclos de venta.

Navision: Permite adelantarse en la economía de e-business de hoy, para cuidar de las relaciones comerciales: con los clientes, prospectos, inversores, proveedores y otros. Ofrece un conjunto integrado de soluciones de administración para la gestión de las relaciones con los clientes, desde compañías que implementan un CRM por primera vez, hasta compañías con una larga experiencia en esta filosofía. Esta solución de CRM crece a medida que el negocio crece.

Salomón: Está diseñado para que el mismo usuario pueda realizar adaptaciones en forma rápida y sencilla sin modificar el código fuente. Al día de hoy Salomón está presente en casi todos los tipos de industrias, ya que han encontrado en este producto un sistema empresarial único, por su capacidad para cubrir los requerimientos particulares de información que van surgiendo conforme la empresa se va desarrollando.

Con sólo oprimir un botón, se puede enviar por fax una carta de agradecimiento a los clientes cada vez que coloquen un pedido. Además, es posible enviar a todos los clientes un correo electrónico confirmando la recepción de sus pedidos. La funcionalidad de Salomón permite dar a los clientes un servicio de excelente calidad. Al acceder la base de datos de Salomón, es posible generar gráficas, reportes analíticos, proyecciones, cartas, etiquetas, solicitudes de pago, confirmación de pedidos o cualquier otro documento que requiera.

mySAP CRM: Proporciona una serie de aplicaciones que ayudan a la empresa a centrarse en el cliente y a aumentar su nivel de eficacia, entre la cuales cabe destacar:

2.2.3 Aplicaciones de CRM

Aplicaciones CRM operativas.

Estas aplicaciones aumentan la capacidad de sus empleados al proporcionarles workplaces basados en roles. Además, permiten la integración perfecta en tiempo real de la interacción front-office y del suministro back-office, al tiempo que sincronizan las interacciones con el cliente a través de todos los canales.

Aplicaciones CRM analíticas.

A partir de su almacén de datos y de otras fuentes, estas aplicaciones le ayudan a comprender lo que quieren sus clientes, así como su comportamiento. También le ayudan a adquirir nuevos clientes y a retener los ya existentes.

Aplicaciones CRM cooperativas .

Estas aplicaciones le ayudan a trabajar más estrechamente con sus clientes. Hoy en día, las empresas pueden elegir entre decenas de soluciones de administración de las relaciones con el cliente. Como información adicional, a continuación se muestran los estimados de participación de mercado para 2002 por ingresos de los cinco principales vendedores de nuevas licencias de CRM:

Tabla N°1 Participación en el mercado.

Compañía	Participación de Mercado 2002 (%)	Participación de Mercado 2001 (%)
Siebel	24.9	28.5
SAP	15.9	10.9
PeopleSoft	4.3	3.9
Oracle	4.3	5.5
Amdocs (Clarificar)	3.2	3.8

Fuente: Gartner Dataquest (junio 2003)

2.2.4 Cómo aprovechar las ventajas de la comunicación electrónica

La Web, e-mail y otros canales electrónicos (como Call Centers) pusieron información, antes difícil de obtener, al alcance de la mano del cliente, lo que hizo que esté mejor informado y en consecuencia, se haya tornado más crítico y poderoso.

Las mencionadas vías de comunicación también hicieron que les resultase más sencillo ponerse en contacto con las compañías, las cuales necesitan ahora responder eficientemente a esta demanda de atención.

Para responder a las necesidades de sus clientes de forma eficaz, las compañías utilizarán CRM para reunir y analizar información sobre ellos, y posteriormente distinguir sus preferencias. Luego, emplearán el producto resultante para el beneficio de ambas partes, lo que las conducirá a establecer relaciones únicas con ellos. Este proceso no sólo requiere la implementación de la nueva tecnología, sino, fundamentalmente, un cambio en la forma de relacionarse con el cliente: es necesario hablar menos y escuchar más, y modificar procesos, por ejemplo, poniendo a prueba las ofertas de mercadotecnia y definiéndolas de acuerdo a las necesidades del cliente.

CRM brinda a la compañía la valiosa oportunidad de conocer al cliente y, por ende, aprender a servirlo. No debe ser desaprovechada.

2.3 METODOLOGÍA XP (PROGRAMACIÓN EXTREMA)

En los tiempos en que la informática empezó a hacerse popular, en el que sólo había pantallas de texto, no había entornos de ventanas y las impresoras usaban papel con agujeros a los lados y rayas horizontales, las aplicaciones eran bastante más sencillas que las actuales. Bastaba uno o dos programadores, que totalmente a su aire y en un plazo razonable de tiempo eran capaces de hacer programas útiles. Para imprimir una lista de clientes, bastaba pulsar en el menú el "1" de imprimir, luego el "3" de clientes y echarle un ojo a la impresora para ver si imprimía o no.

El programador era una especie de "mago" que hacía su aplicación y sólo él sabía cómo funcionaba. Si acaso, y para cubrir el expediente, después de hacer el programa hacía un "flujograma" de los antiguos, no había UML, orientación a objetos ni cosas por el estilo.

Con el tiempo, las aplicaciones fueron siendo cada vez más exigentes, los ordenadores disponían de más memoria y aparecieron los entornos de ventanas. Ahora para imprimir la lista de clientes, había que arrastrar el icono del "señor gordo con bigote" sobre el icono de la impresora, haciendo que caminara "como un egipcio" mientras lo arrastrábamos con el ratón (otro invento nuevo).

La programación extrema es una metodología reciente (alrededor de 5 años) utilizada en el desarrollo de software.

La filosofía de X.P es satisfacer al completo las necesidades del cliente, por eso, lo integra como una parte más del equipo de desarrollo.

2.3.1 Fundamentos de programación extrema

En la programación extrema se da por supuesto que es imposible prever todo antes de empezar a codificar. Es imposible capturar todos los requisitos del sistema, saber qué es todo lo que tiene que hacer ni hacer un diseño correcto al principio.

Es bastante normal hacer un diseño, ponerse a codificar, ver que hay faltantes o errores en el diseño, empezar a codificar fuera del diseño y al final el código y el diseño, o no se parecen, o hemos echado un montón de tiempo en cambiar la documentación de diseño para que se parezca al código.

En vez de tratar de luchar contra todo eso, lo asume y busca una forma de trabajar que se adapte fácilmente a esas circunstancias. Básicamente la idea de la programación extrema consiste en trabajar estrechamente con el cliente, haciéndole mini-versiones con mucha frecuencia (cada dos semanas). En cada mini-versión se debe hacer el mínimo de código y lo más simple posible para que funcione correctamente. El diseño se hace sobre la marcha, haciendo un mini-diseño para la primera mini-versión y luego modificándolo en las siguientes mini-versiones. Además, no hay que hacer una documentación para el diseño, no hay mejor documentación que el mismo código. El código, por tanto, también se modifica continuamente de mini-versión en mini-versión, añadiéndole funcionalidad y extrayendo sus partes comunes.

2.3.2 Los pasos a seguir en un proyecto

En un proyecto usando programación extrema se siguen más o menos los siguientes pasos:

El cliente junto al equipo de desarrollo definen qué es lo que se quiere hacer. Para ello utilizan las "historias de usuario". Una historia de usuario es un texto de una o dos frases en las que se dice algo que debe hacer el sistema.

Es más extensa que un requisito (que suele ser una frase corta) y menos que un caso de uso (que puede ser de una o dos páginas). Se evalúa para cada historia de usuario el tiempo que puede llevar, que debe ser corto, de aproximadamente una semana. Un programador puede estimar con cierta fiabilidad un trabajo que le lleve unos días, pero la estimación es menos fiable si es de un plazo superior a una semana.

Si es más largo, hay que partir la historia en otras más pequeñas. Luego se ordenan en el orden en que se van a desarrollar y se establecen las mini-versiones, de forma que cada mini-versión implementa varias de las historias de usuario.

2.3.3 Características

Las características fundamentales del método programación extrema son:

- **Desarrollo iterativo e incremental:** pequeñas mejoras, unas tras otras.
- **Pruebas unitarias continuas:** frecuentemente repetidas y automatizadas, incluyendo pruebas de regresión. Se aconseja escribir el código de la prueba antes de la codificación. Véase, por ejemplo, las herramientas de prueba JUnit orientada a Java, DUnit orientada a Delphi y NUnit para la plataforma.NET. Estas dos últimas inspiradas en JUnit.
- **Programación en parejas:** se recomienda que las tareas de desarrollo se lleven a cabo por dos personas en un mismo puesto. Se supone que la mayor calidad del código escrito de esta manera -el código es revisado y discutido mientras se escribe- es más importante que la posible pérdida de productividad inmediata.

- Frecuente integración del equipo de programación con el cliente o usuario. Se recomienda que un representante del cliente trabaje junto al equipo de desarrollo.
- **Corrección de todos los errores** antes de añadir nueva funcionalidad. Hacer entregas frecuentes.
- **Refactorización del código**, es decir, reescribir ciertas partes del código para aumentar su legibilidad y mantenibilidad pero sin modificar su comportamiento. Las pruebas han de garantizar que en la refactorización no se ha introducido ningún fallo.
- **Propiedad del código compartida**: en vez de dividir la responsabilidad en el desarrollo de cada módulo en grupos de trabajo distintos, este método promueve el que todo el personal pueda corregir y extender cualquier parte del proyecto. Las frecuentes pruebas de regresión garantizan que los posibles errores serán detectados.
- **Simplicidad** en el código: es la mejor manera de que las cosas funcionen. Cuando todo funcione se podrá añadir funcionalidad si es necesario. La programación extrema apuesta que es más sencillo hacer algo simple y tener un poco de trabajo extra para cambiarlo si se requiere, que realizar algo complicado y quizás nunca utilizarlo.

La simplicidad y la comunicación son extraordinariamente complementarias. Con más comunicación resulta más fácil identificar qué se debe y qué no se debe hacer. Mientras más simple es el sistema, menos tendrá que comunicar sobre este, lo que lleva a una comunicación más completa, especialmente si se puede reducir el equipo de programadores.

Fases de XP

1ª Fase: Planificación del proyecto.

Historias de usuario: El primer paso de cualquier proyecto que siga la metodología X.P es definir las historias de usuario con el cliente.

Las historias de usuario tienen la misma finalidad que los casos de uso pero con algunas diferencias:

Constan de 3 ó 4 líneas escritas por el cliente en un lenguaje no técnico sin hacer mucho hincapié en los detalles; no se debe hablar ni de posibles algoritmos para su implementación ni de diseños de base de datos adecuados, etc.

Son utilizadas para estimar tiempos de desarrollo de la parte de la aplicación que describen. También se utilizan en la fase de pruebas, para verificar si la aplicación cumple con lo que especifica la historia de usuario.

Cuando llega la hora de implementar una historia de usuario, el cliente y los desarrolladores se reúnen para concretar y detallar lo que tiene que hacer dicha historia.

El tiempo de desarrollo ideal para una historia de usuario es entre 1 y 3 semanas.

Release planning: Tras definir las historias de usuario es necesario crear un plan de publicaciones, en inglés "Release planning", donde se indiquen las historias de usuario que se implementarán para cada versión de la aplicación y las fechas en las que se publicarán dichas versiones.

Un "Release planning" es una planificación donde los desarrolladores y clientes establecen los tiempos de implementación ideales de las historias de usuario, la prioridad con la que serán implementadas y las historias de usuario que serán implementadas en cada versión del programa.

Release planning (El alivio planificando)

Después de un "Release planning" tienen que estar claros estos cuatro factores: los objetivos que se deben cumplir (que son principalmente las historias que se deben desarrollar en cada versión), el tiempo que tardarán en desarrollarse y publicarse las versiones de la aplicación, el número de personas que trabajarán en el desarrollo y cómo se evaluará la calidad del trabajo realizado.

Iteraciones. Todo proyecto que siga la metodología X.P se ha de dividir en iteraciones de aproximadamente 3 semanas de duración. Al comienzo de cada iteración los clientes deben seleccionar las historias de usuario definidas en el "Release planning" que serán implementadas. También se seleccionan las historias de usuario que no pasaron el test de aceptación que se realizó al terminar la iteración anterior.

Estas historias de usuario son divididas en tareas de entre 1 y 3 días de duración cada una y serán asignadas a los programadores. Velocidad del proyecto. La velocidad del proyecto es una medida que representa la rapidez con la que se desarrolla el mismo; estimarla es muy sencillo: basta con contar el número de historias de usuario que se pueden implementar en una iteración; de esta forma, se sabrá el cupo de historias que se pueden desarrollar en las distintas iteraciones.

Con la velocidad del proyecto controlaremos si todas las tareas se pueden desarrollar en el tiempo dispuesto para la iteración. Es conveniente reevaluar esta medida cada 3 ó 4 iteraciones y si se aprecia que no es adecuada hay que negociar con el cliente un nuevo "Release Planning".

Programación por pares. La metodología X.P aconseja la programación en parejas pues incrementa la productividad y la calidad del software desarrollado. El trabajo en pareja involucra a dos programadores trabajando en el mismo equipo;

Mientras uno codifica haciendo hincapié en la calidad de la función o método que está implementando, el otro analiza si ese método o función es adecuado y está bien diseñado. De esta forma se consigue un código y diseño con gran calidad.

Reuniones diarias. Es necesario que los desarrolladores se reúnan diariamente y expongan sus problemas, soluciones e ideas de forma conjunta. Las reuniones tienen que ser fluidas y todo el mundo debe tener voz y voto.

2ª Fase: Diseño.

Diseños simples: La metodología XP. sugiere que hay que conseguir diseños simples y sencillos.

Hay que procurar hacerlo todo lo menos complicado posible para conseguir un diseño fácil de entender e implementar que a la larga costará menos tiempo y esfuerzo desarrollar.

Glosarios de términos: Usar una correcta especificación de los nombres de clases, métodos y propiedades ayudará a comprender el diseño y facilitará futuras ampliaciones y la reutilización del código.

Riesgos: Si surgen problemas potenciales durante el diseño, XP. sugiere utilizar una pareja de desarrolladores para que investiguen y reduzcan al máximo el riesgo que supone ese problema.

Funcionalidad extra: Nunca se debe añadir funcionalidad extra al programa aunque se piense que en un futuro será utilizada. Sólo el 10% de la misma es utilizada lo que demuestra que el desarrollo de funcionalidad extra es un desperdicio de tiempo y recursos.

Refactorizar. Consiste en mejorar el código modificando su estructura sin alterar su funcionalidad. Refactorizar supone revisar de nuevo la codificación para procurar optimizar su funcionamiento.

Es muy común reutilizar código ya creado que suele contener funcionalidades que no serán utilizadas y diseños obsoletos; esto es un grave error porque puede generar código inestable y mal diseñado; por este motivo, es necesario refactorizar cuando se va a reutilizar código.

Tarjetas C.R.C. El uso de las tarjetas C.R.C (Class, Responsibilities and Collaboration) permiten al programador centrarse y apreciar el desarrollo orientado a objetos olvidándose de los malos hábitos de la programación procedural clásica. Las tarjetas C.R.C representan objetos; la clase a la que pertenece el objeto se puede escribir en la parte de arriba de la tarjeta, en una columna a la izquierda se pueden escribir las responsabilidades u objetivos que debe cumplir el objeto y a la derecha, las clases que colaboran con cada responsabilidad.

3ª Fase: Codificación.

Como ya se dijo en la introducción, el cliente es una parte más del equipo de desarrollo; su presencia es indispensable en las distintas fases de X.P. A la hora de codificar una historia de usuario su presencia es aún más necesaria.

No olvidemos que los clientes son los que crean las historias de usuario y negocian los tiempos en los que serán implementadas.

Antes del desarrollo de cada historia de usuario el cliente debe especificar detalladamente lo que ésta hará y también tendrá que estar presente cuando se realicen los test que verifiquen que la historia implementada cumple la funcionalidad especificada.

La codificación debe hacerse atendiendo a estándares y patrones de codificación ya creados.

Programar bajo estándares mantiene el código consistente y facilita su comprensión y la escalabilidad.

Crear test que prueben el funcionamiento de los distintos códigos implementados nos ayudará a desarrollar dicho código. Crear estos test antes nos ayuda a saber qué es exactamente lo que tiene que hacer el código a implementar y sabremos que una vez implementado pasará dichos test sin problemas ya que dicho código ha sido diseñado para ese fin. Se puede dividir la funcionalidad que debe cumplir una tarea a programar en pequeñas unidades, de esta forma se crearán primero los test para cada unidad y a continuación se desarrollará dicha unidad, así poco a poco conseguiremos un desarrollo que cumpla todos los requisitos especificados. Como ya mencionamos, X.P opta por la programación en pareja ya que permite un código más eficiente y con una gran calidad.

X.P sugiere un modelo de trabajo usando repositorios de código dónde las parejas de programadores publican cada pocas horas sus códigos implementados y corregidos junto a los test que deben pasar. De esta forma el resto de programadores que necesiten códigos ajenos trabajarán siempre con las últimas versiones. Para mantener un código consistente, publicar un código en un repositorio es una acción exclusiva para cada pareja de programadores.

X.P también propone un modelo de desarrollo colectivo en el que todos los programadores están implicados en todas las tareas; cualquiera puede modificar o ampliar una clase o método de otro programador si es necesario y subirla al repositorio de código. El permitir al resto de los programadores modificar códigos que no son suyos no supone ningún riesgo ya que para que un código pueda ser publicado en el repositorio tiene que pasar los test de funcionamiento definidos para el mismo.

La optimización del código siempre se debe dejar para el final. Hay que hacer que funcione y que sea correcto, más tarde se puede optimizar.

X.P afirma que la mayoría de los proyectos que necesiten más tiempo extra que el planificado para ser finalizados no podrán ser terminados a tiempo se haga lo que se haga, aunque se añadan más desarrolladores y se incrementen los recursos. La solución que plantea X.P es realizar un nuevo "Release planning" para concretar los nuevos tiempos de publicación y de velocidad del proyecto.

4ª Fase: Pruebas.

Uno de los pilares de la metodología X.P es el uso de test para comprobar el funcionamiento del código que estamos desarrollando.

El uso de los test en X.P es el siguiente:

Se deben crear los test con frameworks específicos para tests (JUnit, por ejemplo para Java).

Hay que someter a test las distintas clases del sistema omitiendo los métodos más triviales.

Se deben crear los test que se aplicarán a una clase/método antes de implementarla; en el apartado anterior se explicó la importancia de crear antes los test que el código.

Un punto importante es crear test que no tengan ninguna dependencia del código que en un futuro evaluará. Hay que crear los test abstrayéndose del futuro código, de esta forma aseguraremos la independencia del test respecto al código que evalúa.

Como se comentó anteriormente los distintos test se deben subir al repositorio de código acompañados del código que verifican. Ningún código puede ser publicado en el repositorio sin que haya pasado su test de funcionamiento, de esta forma, aseguramos el uso colectivo del código (explicado en el apartado anterior).

El uso de los test es adecuado para observar la refactorización. Los test permiten verificar que un cambio en la estructura de un código no tiene porqué cambiar su funcionamiento. Para finalizar, un nuevo concepto.

Test de aceptación:

Los test mencionados anteriormente sirven para evaluar las distintas tareas en las que ha sido dividida una historia de usuario.

Para asegurar el funcionamiento final de una determinada historia de usuario se deben crear "Test de aceptación"; estos test son creados y usados por los clientes para comprobar que las distintas historias de usuario cumplen su cometido.

2.4 USOS Y APLICACIONES DE XP

- Extreme Programming se usa actualmente para la creación y desarrollo práctico de software.
- Este se ha usado mucho últimamente, ya que es una metodología ágil para desarrollar software, antes de dar ejemplos de empresas que aplican Extreme programming, citaré las ventajas y desventajas que este tipo de metodología ágil aporta.

2.5 VENTAJAS Y DESVENTAJAS DE EXTREME PROGRAMMING

Ventajas:

- Programación organizada.
- Menor tasa de errores.
- Satisfacción del programador.

Desventajas:

- Es recomendable emplearlo solo en proyectos a corto plazo.
- Altas comisiones en caso de fallar.

2.6 TECNOLOGÍA

2.6.1 Definiciones

La tecnología es el conjunto de teorías y técnicas que permiten el aprovechamiento práctico del conocimiento científico que abarca un conjunto de técnicas, conocimientos y procesos, que sirven para el diseño y construcción de objetos para satisfacer necesidades humanas.

Red

Una red es un número indeterminado de máquinas o computadoras que se comunican entre sí. El cual daría la existencia de conexión con otras redes, a esto lo denominaron subredes.

Una red de computadoras conectadas entre sí, se la utilizan con cierto fin o propósito y describiremos la utilidad de una red.

Compartir recursos, especialmente información con los docentes con relación a las notas de los estudiantes.

Compartir los datos de pagos realizados de los clientes en contabilidad

La información para los estudiantes en la red.

La clasificación en términos de separación física entre nodos como ser, Las redes de área local (LAN) y las áreas de redes amplia (WAN)

Internet

El Internet es una red de millones de computadoras interconectadas por un conjunto de protocolos TCP/IP. Esta tecnología también se usa para extensiones privadas como públicas, también es un conjunto de ordenadores, o servidores, conectados en una red mundial que comparten una misma red de protocolos de comunicación, el cual presta servicios a los ordenadores que se conectan a esa red.

Sitio Web

Web Site o sitio Web, la colección de archivos creados para Internet (páginas gráficas o multimedia) se pueden acceder al sitio a través de su dirección de URL, de donde se puede acceder a las páginas y archivos.

Un sitio Web no necesariamente debe localizarse en el sistema de cómputo. Los documentos que integran el sitio Web pueden ser localizados en diferentes lugares sin importar la distancia, puede ser de un país a otro es decir no existen fronteras, este equipo de cómputo o servidor Web puede tener más de un sitio Web y atender de continuo a visitantes de cada uno de los diferentes sitios.

2.7 HERRAMIENTAS

Apache

Es un servidor más utilizado para el Internet, Apache es un sistema útil en Unix, Linux, para lo que es en Windows existen un emulador, aunque este emulador no se le considere tan robusto como en Apache de Unix.

Definiciones de Apache en la Web:

Es el servidor http más ampliamente disponible en Internet. Soportando las diferentes aplicaciones de lenguajes de PHP y PERL.

Servidor de páginas Web

El servidor libre de la Web como el Apache tiene gran cabida en el mercado internacional, precisamente por ser un software libre sin ningún costo.

Flash

Adobe Flash (Macromedia Flash) se refiere tanto al programa de edición multimedia como a Macromedia Flash Player, escrito y distribuido por Adobe, que utiliza gráficos vectoriales e imágenes ráster, sonido, código de programa, flujo de vídeo y audio bidireccional.

En sentido estricto, Flash es el entorno y Flash Player es el programa de máquina virtual utilizado para ejecutar los archivos generados con Flash.

Los archivos de Flash, que tienen generalmente la extensión de archivo SWF, pueden aparecer en una página Web para ser vista en un navegador, o pueden ser reproducidos independientemente por un reproductor Flash. Los archivos de Flash aparecen muy a menudo como animaciones en páginas Web y sitios Web multimedia, y más recientemente Aplicaciones de Internet Ricas. Son también ampliamente utilizados en anuncios de la Web.

En versiones recientes, Macromedia ha ampliado Flash más allá de las animaciones simples, convirtiéndolo en una herramienta de desarrollo completa, para crear principalmente elementos multimedia e interactivos para Internet.

PHP

(PHP Hypertext Pre-processor). Lenguaje de programación usado generalmente en la creación de contenidos para sitios web. Es un lenguaje interpretado, especialmente usado para crear contenido dinámico web y aplicaciones para servidores, aunque también es posible crear aplicaciones gráficas utilizando la biblioteca.

Generalmente los Scripts en PHP se embeben en otros códigos como HTML, ampliando las posibilidades del diseñador de páginas web enormemente. La interpretación y ejecución de los Scripts PHP se hacen en el servidor, el cliente (un navegador que pide una página web) sólo recibe el resultado de la ejecución y jamás ve el código PHP.

Permite la conexión a todo tipo de servidores de base de datos con MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite. PHP es una alternativa a otros sistemas como el ASP.NET/C#/VB.NET de Microsoft o a ColdFusion de Macromedia, a JSP/Java de Sun Microsystems,

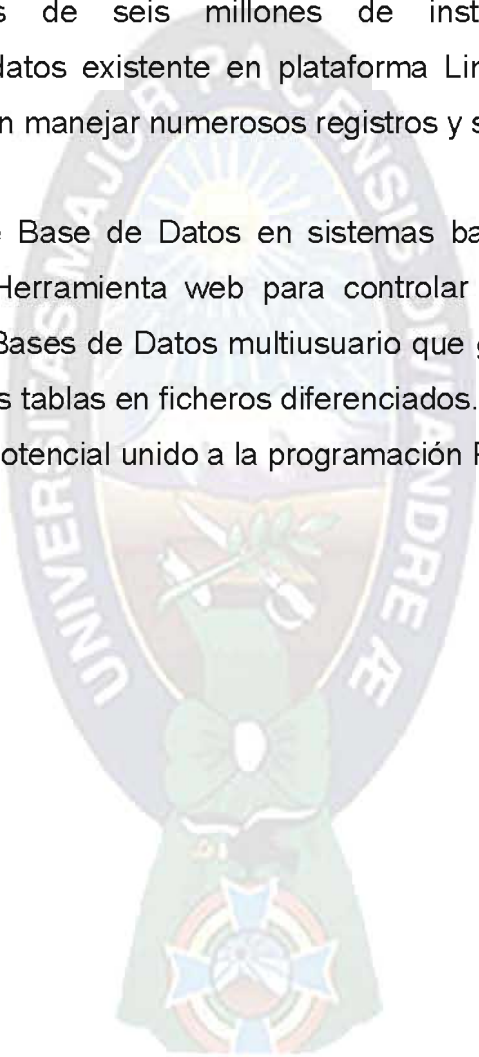
La ventaja con los de Microsoft o Macromedia es que es totalmente gratuito, no hay que pagar licencias.

MySql

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. Donde los servidor de Bases de datos existente en plataforma Linux, recomendable para desarrollos que necesiten manejar numerosos registros y sesiones.

Plataforma Standard de Base de Datos en sistemas basados en Unix (Linux / BSD). PhpMyAdmin - Herramienta web para controlar y manejar su base de MySql es un gestor de Bases de Datos multiusuario que gestiona bases de datos relacionales poniendo las tablas en ficheros diferenciados.

Cabe destacar su gran potencial unido a la programación PHP.



CAPITULO 3

3 MARCO APLICATIVO

En este capítulo se presenta la aplicación del método programación extrema (XP) y la herramienta para el correspondiente diseño de implementación.

3.1 Programación Extrema (XP)

1ª Fases: planificaron del proyecto

En esta se establecen las historias de usuarios, se define prioridades de y se expone el plan de trabajo a entregar.

Para el desarrollo óptimo del sistema CRM (Customer Relationship Management), se conforma el equipo de trabajo que se describe a continuación.

Tabla N° 2 Equipo de Proyecto

Descripción	Responsable	Rol
Cliente	Administrador de Lengua_udabol	Establece las historias de usuarios
Jefe de proyecto	Desarrollador del sistema	Coordina la historia de usuarios con cliente
Desarrollador	Una sola pareja por desarrollador y un coordinador de información del centro de enseñanza	Coordina el desarrollo del software

a) Historia de usuario:

A medida que transcurre el proyecto se establecen las siguientes historias de usuarios.

- a) plan de entrega.- Se estableció plan de publicaciones (release planning) donde se indica los objetivos, el tiempo de implementación y el método de evaluación o test de aceptación.

HISTORIA DE USUARIO	N° 1	C.R.M.
Nombre: Llenado de formulario a clientes nuevos		Prioridad
Historia: El sistema anterior que registraba los datos de los clientes en formularios (hojas) serán migrados al sistema		
Objetivo: Desarrollar rutinas donde el sistema genere el seguimiento a todo cliente registrado inscrito		
Estimación.		Dependencia

HISTORIA DE USUARIO	N° 2	C.R.M.
Nombre: Inscripción		Prioridad
Historia: La inscripción a cada curso de los módulos por bimestre debe ser guardado para formularios de dichos cursos		
Objetivo: Mostrar en cada paralelo materias y docentes donde están inscritos, todo esto se hará por medio del sistema C.R.M.		
Estimación.		Dependencia

HISTORIA DE USUARIO	N° 3	C.R.M.
Nombre: Pago de materias		Prioridad
<p>Historia:</p> <p>Los pagos que se realizan en otra unidad exclusiva pero el estudiante necesita saber los módulos pagados por tanto debe estar incluido dentro el sistema</p>		
<p>Objetivo:</p> <p>Los pagos realizados de las materias se controlara por medio de las facturas , anotando en el sistema el numero de factura</p>		
Estimación.		Dependencia

HISTORIA DE USUARIO	N° 4	C.R.M.
Nombre: Horarios		Prioridad
<p>Historia:</p> <p>El estudiante cada principio de cada bimestre necesita saber los horarios paralelos y docentes a las que se inscribió</p>		
<p>Objetivo:</p> <p>Hacer horarios que puedan indicar tanto los horarios de docentes materias y a los módulos que corresponden.</p>		
Estimación.		Dependencia

HISTORIA DE USUARIO	N° 5	C.R.M.
Nombre: Moras		Prioridad
<p>Historia:</p> <p>El retraso o abandono de materias del bimestre resulta en pago de moras, donde no existe informe de estos pagos realizados por los estudiantes, ni la cantidad de moras pagadas ni los retrasos.</p>		
<p>Objetivo:</p> <p>Realizar control de las moras pagadas por los estudiantes y las moras que debe cada estudiante</p>		
Estimación		Dependencia

HISTORIA DE USUARIO	N° 6	C.R.M.
Nombre: Generación de reportes		Prioridad
<p>Historia:</p> <p>El estudiante quiere saber todo lo referente con el aspecto académico materias aprobadas que esta cursando y otros</p>		
<p>Objetivo:</p> <p>Generar reportes del estado del estudiante con lo que envuelve con lo académico.</p>		
Estimación.		Dependencia

HISTORIA DE USUARIO	N° 7	C.R.M.
Nombre: Migración de datos		Prioridad
<p>Historia:</p> <p>Tanto los datos ingresados de los clientes deben ser utilizados y guardados lo mismo que el aspecto de las notas módulos y otros.</p>		
<p>Objetivo:</p> <p>Migrar esos reportes a un gestor de base de datos (MySQL) el cual este gestor servirá para mostrar en la Web</p>		
Estimación.		Dependencia

b) Release Planning

Como pide el método XP la aplicación de release planning es uno de los pasos mas importantes de programación extrema.

En esta fase se define prioridades por el número de historias de usuarios, el plan de publicaciones queda de la siguiente manera.

RELEASE PLANNING(Historia de Usuario N° 1)	
Objetivo	Desarrollar rutinas donde el sistema genere el seguimiento a todo cliente registrado inscrito
Tiempo de implementación	2 Semanas
Personas asignadas	Una sola pareja compuesta por el desarrollador y el coordinador de informaciones de todo el centro del idioma Ingles
Método de evaluación	Verificado por un buscador de pagina Web en el computador del cliente.
Test de aceptación	ACEPTADO

RELEASE PLANNING(Historia de Usuario N° 2)	
Objetivo	Mostrar en cada paralelo materias y docentes donde están inscritos , todo esto se hará por medio del sistema C.R.M.
Tiempo de implementación	2 Semanas
Personas asignadas	Una sola pareja compuesta por el desarrollador y el coordinador de informaciones de todo el centro del idioma Ingles
Método de evaluación	Por el estudiante determinando por la vía Web
Test de aceptación	ACEPTADO

RELEASE PLANNING(Historia de Usuario N° 3)

Objetivo	Los pagos realizados de las materias se controlara por medio de las facturas , anotando en el sistema el numero de factura
Tiempo de implementación	1 Semana
Personas asignadas	Una sola pareja compuesta por el desarrollador y el coordinador de informaciones de todo el centro del idioma Ingles
Método de evaluación	Por los encargados del manejo económico las cuales verificaran a la ves los estudiantes
Test de aceptación	ACEPTADO

RELEASE PLANNING(Historia de Usuario N° 4)

Objetivo	Hacer horarios que puedan indicar tanto los horarios de docentes materias y a los módulos que corresponden.
Tiempo de implementación	1 Semana
Personas asignadas	Una sola pareja compuesta por el desarrollador y el coordinador de informaciones de todo el centro del idioma Ingles
Método de evaluación	Por los docentes y los estudiantes por medio de la vía Web
Test de aceptación	ACEPTADO

RELEASE PLANNING(Historia de Usuario N° 5)

Objetivo	Realizar control de las moras pagadas por los estudiantes y las moras que debe cada estudiante
Tiempo de implementación	1 Semana
Personas asignadas	Una sola pareja compuesta por el desarrollador y el coordinador de informaciones de todo el centro del idioma Ingles
Método de evaluación	Por el cliente por medio de la pagina Web
Test de aceptación	ACEPTADO

RELEASE PLANNING(Historia de Usuario N° 6)

Objetivo	Generar reportes del estado del estudiante con lo que envuelve con lo académico.
Tiempo de implementación	3 Semanas
Personas asignadas	Una sola pareja compuesta por el desarrollador y el coordinador de informaciones de todo el centro del idioma Ingles
Método de evaluación	Pruebas por generación de reportes, tanto el aspecto académico y logístico.
Test de aceptación	ACEPTADO

RELEASE PLANNING(Historia de Usuario N° 7)

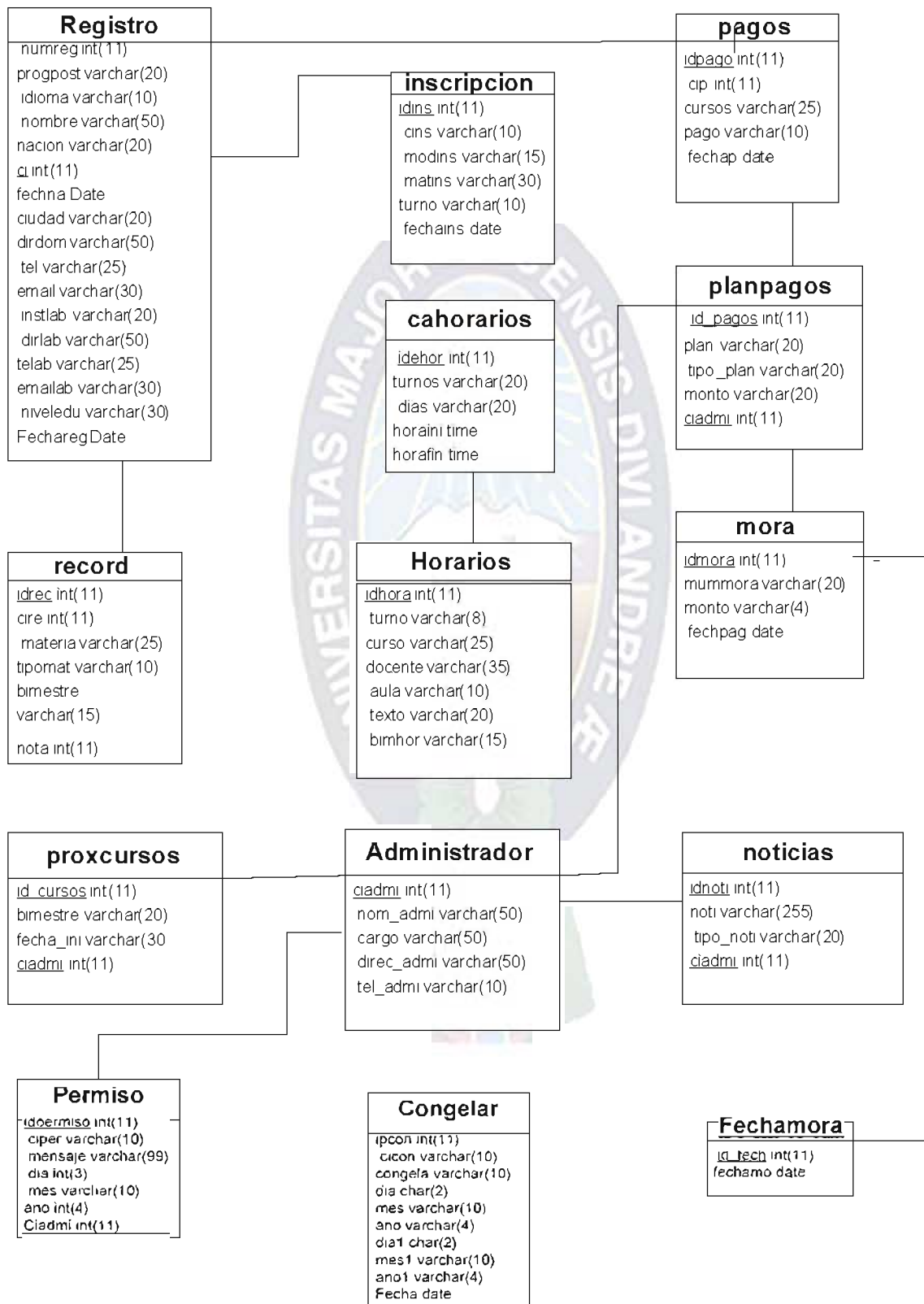
Objetivo	Migrar esos reportes a un gestor de base de datos (MySQL) el cual este gestor servirá para mostrar en la Web
Tiempo de implementación	3 Semanas
Personas asignadas	Una sola pareja compuesta por el desarrollador y el coordinador de informaciones de todo el centro del idioma Ingles
Método de evaluación	Verificar la existencia de los datos de migración mediante un buscador Web en el computador del cliente.
Test de aceptación	ACEPTADO

2.- Fase de diseño.

De acuerdo a la Metodología programación Extrema (XP) existe la posibilidad de expandirse a medida que va aumentando los requerimientos que pide el cliente. En la figura N° 1 se muestra los el estado de la base de datos se desarrollan en el procesador de hipervínculo PHP nos permitirá la implementación de diferentes enlaces y reportes necesarios para la eficiencia del sistema.

Figura N° 3

Estructura de las tablas de la base de datos de Lengua_Udabol



3.2 fase de iteración

Esta fase es la representativa de programación extrema llenado de tarjetas C.R.C. de acuerdo a la historia de usuario, la aplicación de la mayoría en la practica de los valores XP.

Tabla N° 3 Iteración del sistema

Nº de iteración	Historia de Usuarios involucrado	Test aceptado (por historia Usuario)	Test Fallido (por historia Usuario)	Retroalimentación (por historia Usuario)	Nº de semanas
1	HS. 1	HS. 1			2 Semanas
	HS. 2	HS. 2		HS. 2	2 Semanas
2	HS. 3	HS. 3			1 Semana
3	HS. 4	HS. 4		HS. 4	1 Semana
4	HS. 5	HS. 5			1 Semana
5	HS. 6	HS. 6		HS. 6	3 Semanas
6	HS. 7	HS. 7			3 Semanas
Total					13 Semanas

Como la metodología un diseño simple nos permite conseguir un diseño fácil de entender e implementar y así da una idea general del sistema a desarrollar.

Figura N° 4 CRC Correspondiente a la Historia N° 1

Seguimiento	Nª 1
Llenar _ formulario	Motor
Desplegar formulario	PHP

formulario	Nª 1
Llenar _ formulario	Motor
Desplegar formulario	PHP
	Buscar

Figura N° 5 CRC

Corresponde a usuario N° 2

Inscripción	Nª 2
Llenar _ formulario	Motor
	PHP
	Registrar
	Mostrar
	Modificar
	Borrar
	Imprimir

Formulario	Nª 2
Desplegar formulario	Motor
Imprime _ digital	PHP
	Registrar
	Mostrar
	Modificar
	Borrar
	Imprimir

Descripción de los módulos de la tarjetas CRC. historia de usuario N° 2

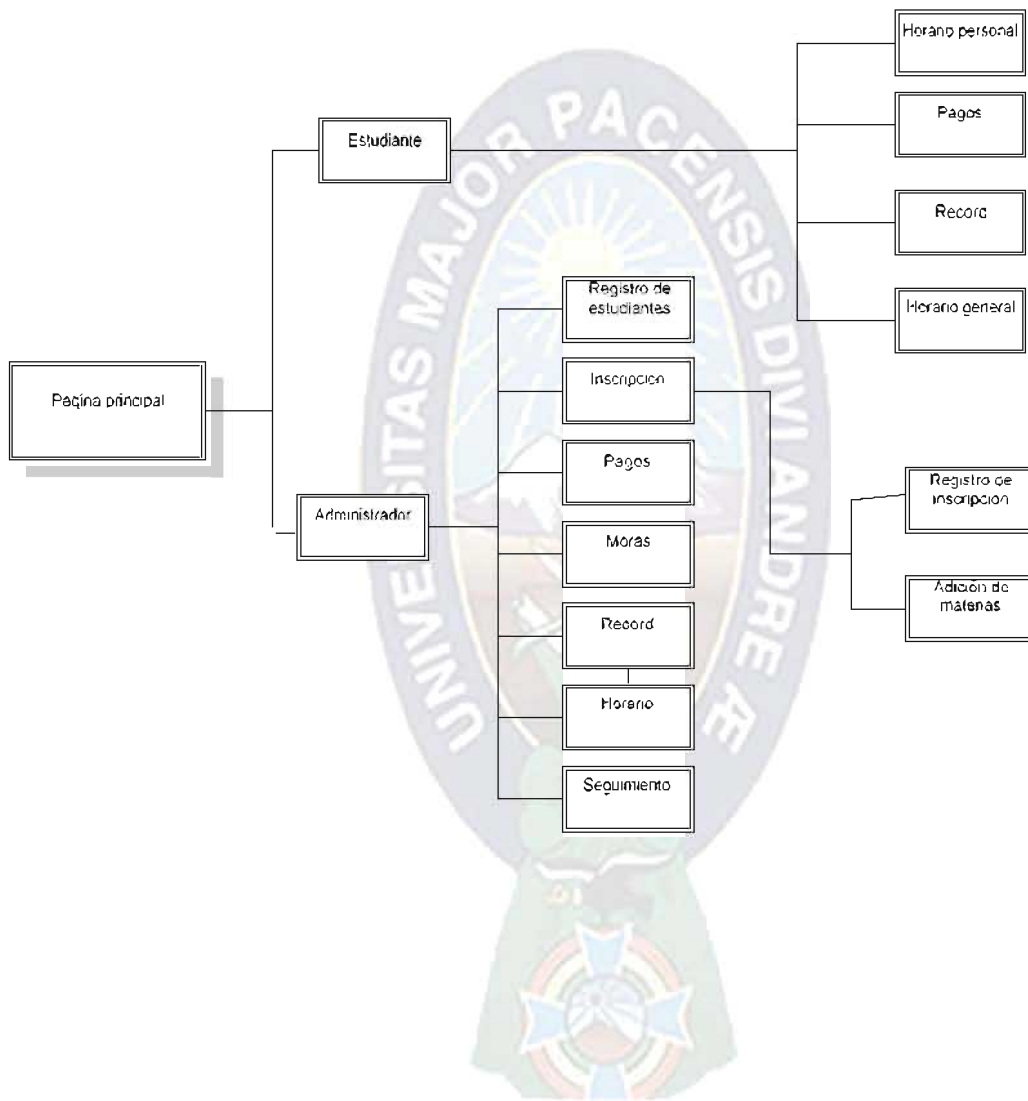
Tabla N° 4 Glosario de términos

TERMINOS	DESCRIPCION
Motor. php	Inicializa la base de datos
Registrar	Realizado por PHP y de asistencia javascript para llenado ordenado de los datos
Mostrar	Una vez llenado la base de datos se muestra lo que se lleno en el registro, donde se verifica si es correcto el llenado de la misma
Modificar	En caso de existir un error en llenado del registro que es un dato importante se pueda modificar los datos de los clientes
Borrar	Esta operación permite borrar los datos equívocos en llenado del formulario dentro de la base de datos.
Imprimir	La existencia de la opción de los formularios como comprobante del registro u otros

2.3 Diagrama navegacional

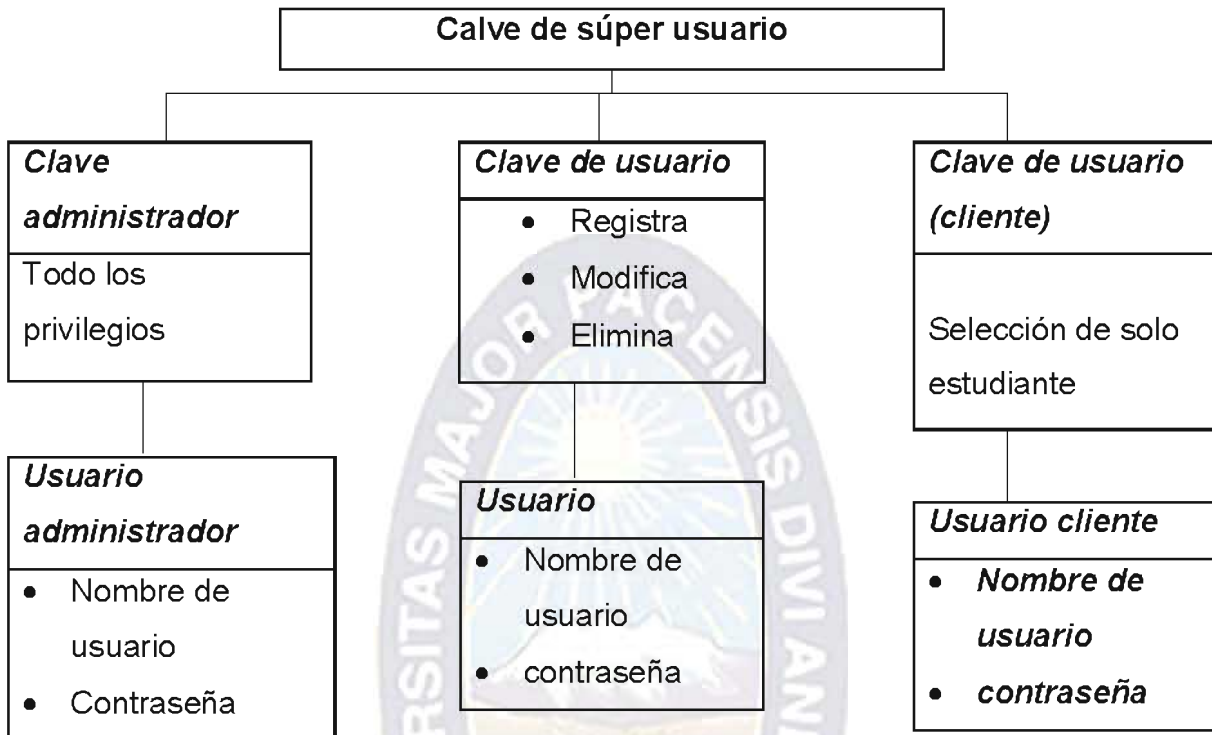
Se muestra el diagrama navegacional de módulos, el cual fue desarrollado las interfaces del sistema y sus correspondientes módulos desplazando en los menús.

Figura N° 6 Diagrama navegacional del sistema



2.4 Esquema de análisis para la gestión de clave

Figura N° 7



3ª Fase de Codificación

La tercera FACE de programación extrema se debe tratar con el cliente por ser parte del equipo de desarrollo.

A la hora de modificar una historia de usuario su presencia es imprescindible por la importancia del resultado esperado.

El proceso como entrada de datos por migración las salidas fueron los módulos creados en PHP.

a) Entrada a fase de generación de código

- Diseño de base de datos la figura N° 1
- Descripción de formato de archivos

b) Salida de generación de código

Modulo desarrollado para la lectura de reportes generados, de acuerdo a la especificación de registro lectura de archivos.

Modulo para inserción de datos de la base de datos

- Abrir conexiones en la base de datos
- Verifica si el registro es valido
- Insertar registro

Proceso de migración de datos

Tablas N° 5 Proceso de migración de datos

Num	Procedimiento	Descripción
1	Consulta de cursos	Se pregunta los cursos que quiere inscribirse los módulos y la aspiración a técnico medio o técnico superior
2	Validar datos	Se realiza la validación de datos de la primera visita que hizo a la institución si esta dentro de los parámetros o alcances de la institución
3	Inserción de datos	Con los datos obtenidos, se inserta a su correspondiente tabla dentro de la base de datos que ya esta direccionado por pregunta.

4ª Fase pruebas

En esta fase se evalúa las pruebas de migración y cargado de la base de datos, los procesos de migración y las salidas de datos dentro de la evaluación de resultados de la migración.

- a) entrada fase de evaluación
- Entorno de migración

- Modulo de lectura y validación de datos
 - Procedimientos de migración
- b) Salida de fase de evaluación los resultados de pruebas de migración
- c) Evaluaron de resultados de pruebas de migración de acuerdo al resultado obtenido dentro de la nueva migración de datos.

Tabla N° 6

Declararon de métodos de evaluación

N° Historia de usuario	Método de evaluación
1	Verificado de la existencia de datos para el seguimiento a cada cliente.
2	Verifica que un cliente verifica se ha inscrito de la forma correcta a un modulo al que corresponda
3	Verifica si pagaron los modulo por materias para el seguimiento normal de avance de cursos
4	Verifica si están inscritos en los horarios y materias de los docentes de manera ordenada
5	Verifica si abandono por algún tiempo cierto modulo o solo una materia, el cual produce el pago de moras por el tiempo de abandono
6	Verifica si da reportes adecuados a cada consulta tanto del estudiante como administrativo
7	Verificar que los datos que se manejan son correctas si vota los datos de los clientes de manera ordenado

Figura N° 8 Test

Análisis de la validación de los test de la historia de usuarios en cada iteración correcta.

Historia usuario	Distribución de test					
7						■
6					■	■
5				■		
4			■	■		
3		■	■			
2	■	■				
1	■					
Iteración	1	2	3	4	5	6

■ Test aceptado

■ Test retroalimentado

■ Test Fallido

En esta figura 6 podemos visualizar que no existieron muchas fallas, solo en la iteración 2 de la historia numero tres esto por la interpretación de la historia 3 y los arreglos posteriores.

Los de color verde son los test de aceptación, los colores anaranjado son los test de retroalimentación, finalmente los de color rojo son los test fallido.

Por tanto los indicadores de test aceptados interpretan la satisfacción de los resultados del proyecto.

Tabla N° 7 Parámetro de medición del sistema

Parámetro de medición	Cuenta		Simple	Medio	Complejo		
Numero de entradas de usuario	10	*	3	4	6	=	60
Numero de salidas de usuario	10	*	4	5	7	=	70
Numero de peticiones de usuario	4	*	3	4	6	=	24
Numero de archivos	11	*	7	10	15	=	165
Numero de interfaces externas	0	*	5	7	10	=	
Total							319

$$Pf = \text{Cuenta total} * [0.65 + 0.01 * (Fi)]$$

Pf = Punto función

Cuenta total = Suma de toda la tabla

Fi = Valores de ajuste de la complejidad

Tabla N° 8 Factores de intervención para medición

Factor	0	1	2	3	4	5
¿Requiere el sistema copias de seguridad y recuperación fiable?						*
¿Requiere comunicación de datos?						*
¿Existen funciones de procedimientos distribuidas?					*	
¿Es crítico el rendimiento?		*				
¿Será utilizado el sistema en un entorno operativo existente y fuertemente utilizado?						*
¿Requiere el sistema de datos interactivos?					*	
¿Requiere el sistema de entrada de datos interactivos que las transacciones de entradas se lleven acabo sobre las múltiples pantallas o variadas operaciones?			*			
¿Se actualiza los archivos maestros de forma interactiva?					*	
¿Son complejas las entradas, salidas, archivo o peticiones?				*		
¿Es complejo el procedimiento interno?					*	
¿Se ha diseñado el código para que sea reutilizable?						*
¿Están incluidos el diseño la conversión y instalación?				*		
¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?						*
¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizables por el cliente?						*

$$\sum F_i = F_1 + F_2 + F_3 + F_4 + F_5 + F_6 + F_7 + F_8 + F_9 + F_{10} + F_{11} + F_{12} + F_{13} + F_{14}$$

$$\sum F_i = 55$$

Entonces el resultado de punto función es:

$$P_f = \text{Cuenta total} * [0.65 + 0.01 * (f_i)]$$

$$P_f = 319 * [0.65 + 0.01 * 55]$$

$$P_f = 382.8$$

Si consideramos F_i como 100% será:

$$P_f = 319 * [0.65 + 0.01 * 70]$$

$$P_f = 430.65$$

En conclusión diremos que $382.8/430.65=88.80$ de funcionalidad del sistema realizado

Esto quiere decir que basándose en el PF obtenido se puede estimar el tamaño global de implementación de las funciones de iteración del sistema

El valor de punto función proporciona una medida de software, esto permite establecer estimaciones objetivas de calidad.

3.5 Calidad de Software

Para medir los software existen muchos métodos de evaluación, uno de estos métricas de medición es **ISO 9126** es un estándar internacional para la evaluación del Software.

Está supervisado por el proyecto SQuaRE, ISO 25000:2005, el cuál sigue los mismos conceptos.

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

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

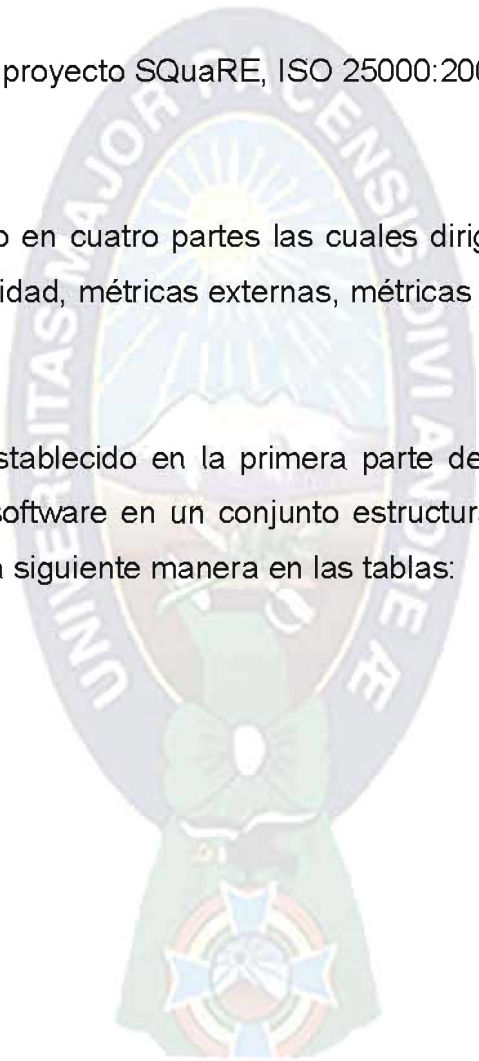


Tabla N° 9 Árbol de requerimiento de calidad

1 Funcionalidad
1.1 Aspecto de búsqueda y recuperación
1.2 Mecanismo de filtrado de información
1.3 Navegabilidad
1.3.1 Orientación
1.3.2 Nivel de desplazamiento
1.4 Dominio orientado al usuario
1.4.1 Relevancia de contenido
2 Usabilidad
2.1 Mapa de sitio
2.2 Mecanismo de ayuda
2.3 Interfaces e estética
2.3.1 Cohesividad agrupar el control de objeto principal
2.3.2 Uniformidad el control de enlace
2.2.3 Uniformidad el estilo global
3 Eficiencia
3.1 Performancia
3.1.1 Página de acceso rápido
3.2 Accesibilidad
3.2.1 Accesibilidad de información
3.2.2 Accesibilidad de ventanas
4 Confiabilidad
4.1 No deficiente
4.1.1 Error de enlaces
4.1.2 Deficiencia

La tabla numero 8 nos muestra criterios elementales de valores para cada atributo de las características de Funcionabilidad.

Tabla N° 10 Resultado de las Preferencias de Funcionabilidad

Código	Atributo	Definición	Criterio elemental	IEi
1.1	Aspecto de búsqueda y recuperación	Búsqueda de algún elemento en particular	Variable normalizada	90
1.2	Mecanismo de filtrado de información	Permite seleccionar una opción para filtrar	Variable normalizada	90
1.3	Navegabilidad	Predecir los contenidos de asociación	Multi nivel discreto	80
1.3.1	Orientación	Idea sobre la calidad de partida de una pagina	Multi nivel discreto	70
1.3.2	Nivel de desplazamiento	Nivel de desplazamiento que navegante de debe realizar (resolución 600*800)	Multi nivel discreto	70

La tabla numero 7 nos muestra criterios elementales de valores para cada atributo de las características de Usabilidad

Tabla N° 11 Resultado de las Preferencias de Usabilidad

Código	Atributo	Definición	Criterio elemental	IEi
2.1	Mapa de sitio	Con componentes gráficos que muestra la arquitectura global (jerárquica) del sitio Web	Criterio discreto y absoluto	70
2.2	Mecanismo de ayuda	Permite conocer al cliente características de la Web	Criterio discreto y absoluto	100
2.3.1	Cohesividad agrupar el control de objeto principal	Indica el agrupamiento adecuado de enlaces	Criterio de referencia de calidad directa	100
2.3.2	Uniformidad el control de enlace	Todo los sitios de enlace son de mismo color	Criterio de referencia de calidad directa	100
2.3.3	Uniformidad el estilo global	Todo la Web es uniforme estilo y fuente también en su totalidad	Criterio de referencia de calidad directa	90

La tabla numero 9 nos muestra criterios elementales de valores para cada atributo de las características de Usabilidad criterios de preferencia elemental.

Tabla N° 12 Resultado de las Preferencias de Eficiencia

Código	Atributo	Definición	Criterio elemental	IEi
3.1.1	Página de acceso rápido	Mide el tamaño de todas las páginas de aplicación considerando todo sus componentes	Multi_variable, contenido	80
3.2.1	Accesibilidad de información	Que se encuentra en la página versión de solo texto e imagen	Multi_nivel	80
3.2.2	Accesibilidad de ventanas	Los frames o marcos organizan a una ventana en diferentes áreas su contenido	Multi_variable, contenido	100

La tabla número 10 nos muestra criterios elementales de valores para cada atributo de las características de Confiabilidad

Tabla N° 13 Resultado de las Preferencias de Confiabilidad

Código	Atributo	Definición	Criterio elemental	IEi
4.1.1	Error de enlaces	Representa los enlaces encontrados que conducen a destinos ausentes y como parámetro X_i =cantidad total de enlace error X_j =cantidad total de enlaces $100-X$ = Resultado	Variable normalizada, continua, absoluta	100
4.1.2	Deficiencia	Si existe mala presentación del sitio Web a los navegadores Página en construcción También páginas muertas	Referencia de claridad directa	80

Por medio de la tabla podemos determinar a cierto grado la eficiencia de los resultados del sistema.

Tabla N° 14 Resultados de la preferencia global

Atributo	Preferencia Global
Funcionabilidad	80
Usabilidad	92
Eficiencia	86.67
Confiabilidad	90

3.6 Análisis de beneficio

Los beneficios por el proyecto de grado realizado para la institución Lingua_Udabol son intangibles tanto económico como los alcances inmediatos por la rapidez de información que brinda el sistema.

- 1.- incremento de velocidad de información para los clientes, después de llenado de formularios los registro serán utilizados para generar reportes del estado del los estudiantes.
- 2.- capacidad en el volumen de información, se podrá almacenar gran cantidad de datos sin ninguna dificultad, permitiendo el manipuleo a favor de la institución y de los clientes
- 3.- El control académico otro aspecto importante el seguimiento de todo lo relacionado con las notas, exámenes, materias aprobadas estará al alcance del estudiante en todo momento.
- 4.- control de procesos. Será otro beneficio importante se publicaran los tramites necesarios para la conclusión de materias y otros ineludibles.

5.- información para la toma de decisión. Los beneficios que otorga el sistema ya sea por los formularios y reportes ayudara a tomar decisiones correctas con antelación, permitiendo la optimización oportuna.



CAPITULO 4

4.1 CONCLUSIONES Y RECOMENDACIONES

En este capítulo se expone la conclusión eficientes del desarrollo del proyecto como también las recomendaciones necesarias para el buen funcionamiento del sistema CRM (Customer Relationship Management).

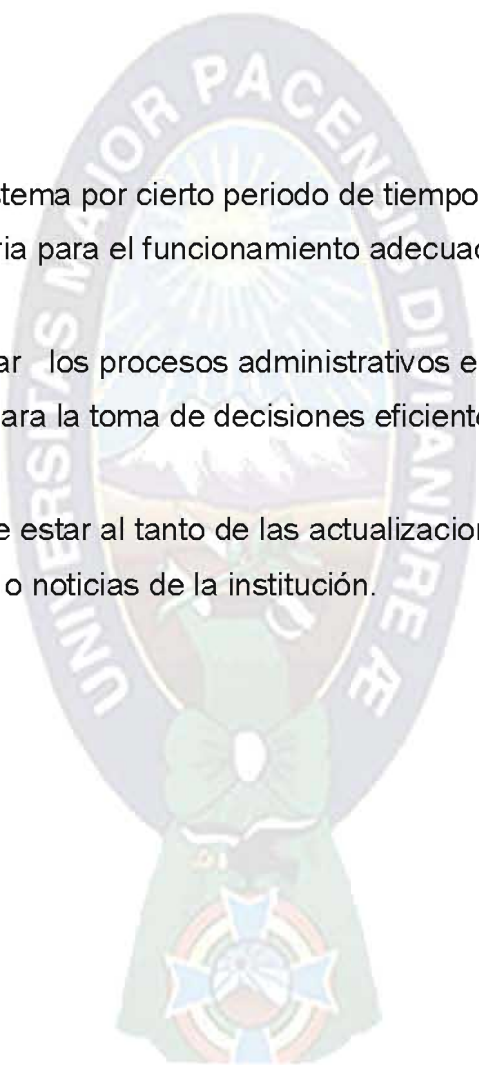
4.1 Conclusiones

- Se logro analizar a profundidad las ventajas y desventajas de la implementación del sistema.
- Por medio del sistema se logra registrar a todo los clientes que pasan por la institución.
- Se logro por medio de la base de datos el resguardo de la información de todos los clientes ya como estudiantes regulares como los que solo se registraron.
- Los datos que se registran en la base de datos no podrán ser modificados por los clientes, solo el director podrá modificar los datos.
- Se proporciono una herramienta que muestra la información de los clientes que pagaron por las materias y las que no están pagadas, también el congelamiento previo a la mora de los pagos.
- Se logro proporcionar por medio de los reportes los horarios las aulas de todas las materias y a los módulos que corresponden.
- Cuando existe retraso de los pagos, los clientes deberán pagar por mes atrasado cierto monto, también se logra controlar esta situación.
- El sistema controla el pago realizado por los módulos con relación a las materias, muestra los montos y las fechas que pagaron.
- El cliente podra ver las materias con los docentes que corresponden y las aulas asignados para cada materia.

- Por medio del sistema se verificara el seguimiento a los clientes o estudiantes, como solicitud de permisos con fechas correspondientes, los motivos de la solicitud del permiso.
- El sistema por estar en la Web, el cliente podrá ver todo con relación a los aspectos académicos.

4.2 Recomendaciones

- La revisión del sistema por cierto periodo de tiempo para que exista la eficiencia necesaria para el funcionamiento adecuado
- Se debe coadyuvar los procesos administrativos en términos de sistematización, para la toma de decisiones eficientes y óptimos
- El estudiante debe estar al tanto de las actualizaciones de los cursos y reportes de notas o noticias de la institución.



REFERENCIAS BIBLIOGRAFICAS

Goldenberg, Barton.

<http://www.tress.com.mx/boletin/Noviembre2002/crm.html>

<http://www.jdedwards.es/public/o.1921.328%257E1162%257E8481.00.html>

http://www.improven-consultores.com/paginas/documentos_gratuitos/que_crm.php/t_blank

http://www.improven-consultores.com/paginas/documentos_gratuitos/realidad_crm.php

http://www.clickz.com/crm/crm_strat/article.php/978061.

<http://www.gestiopolis.com/canales/demarketing/articulos/37/crmcliente.htm>

<http://www.crmguru.com/content/features/sims01.html>

<http://www.liderazgoymercadeo.com>

<http://www.idg.es/computerworld/>

<http://www.mercado.com.ar/altadireccion/>

<http://www.infochannel.com.mx/>

www.gartner.com

<http://www.infochannel.com.mx/>

www.netmedia.info/informationweek/

www.claveempresarial.com/soluciones/sscrm.shtml

www.tecnologiaempresarial.info/

<http://www.maestrosdelweb.com/editorial/phpintro/www.php.net>

Anexo A

Análisis de problema

	PROBLEMA	CAUSA	EFECTO	INVOLUCRADOS
P1.	No tener acceso inmediato a la información de parte de las autoridades administrativas sobre el manejo de todos los clientes.	Desinformación.	Falta de atención inmediata a las necesidades de los clientes.	Administrativos.
P2.	El sistema de seguimiento al cliente que se utiliza todavía no cuenta con la información suficiente para los estudiantes y administrativas.	El sistema no cubre las necesidades de demandada.	No se da buena atención al cliente	Administrativos y estudiantes.
P3.	La inseguridad de manipuleo de los datos en el sistema utilizado.	Un sistema muy censillo y que no maneja seguridad.	Inseguridad de datos.	Administrativos y estudiantes de la institución.
P4.	Consultas frecuente de las notas por modulo de los estudiantes.	No saber las materias a cursar y pagar.	Pagos de módulo no correspondiente o no saber cual pagar.	Clientes, caja y personal.

P5.	Comunicación directa con Caja.	Pagos inapropiados de los estudiantes.	Confusión en caja y reclamos.	Personal de caja y empleados.
P6.	Las materias pagadas por los diferentes módulos.	La existencia de registro adecuado del control de pagos por módulos.	Confusión para los docentes a la hora de tomar exámenes.	Los docentes y estudiantes.
P7.	Abandono de módulos.	Saber si reprobó o no pago.	Déficit y cierre de materias.	Los clientes y la institución.
P8.	Control de pagos de moras.	Falta de control de moras pagadas.	Control inapropiado.	Los administradores.
P9.	Si se dio seguimiento al cliente desde el inicio.	La no existencia de un sistema de seguimiento.	Atención adecuada.	El centro de enseñanza.
P10.	Consulta frecuente de los horarios y aulas.	El estudiante no tiene acceso directo al sistema.	Desinformación del estudiante.	Los estudiantes, docentes y administradores.



Anexo B

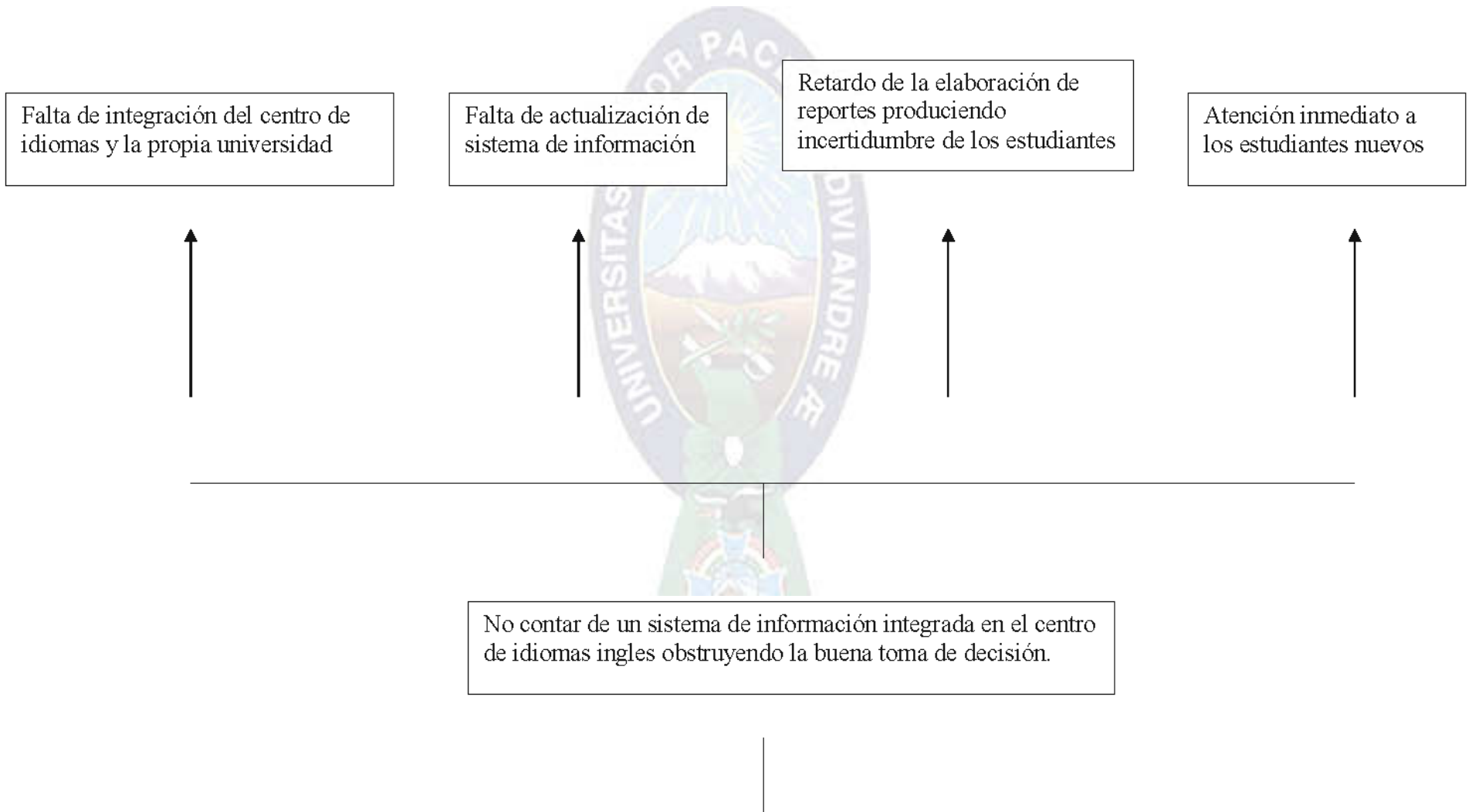
Matriz del Marco Lógico

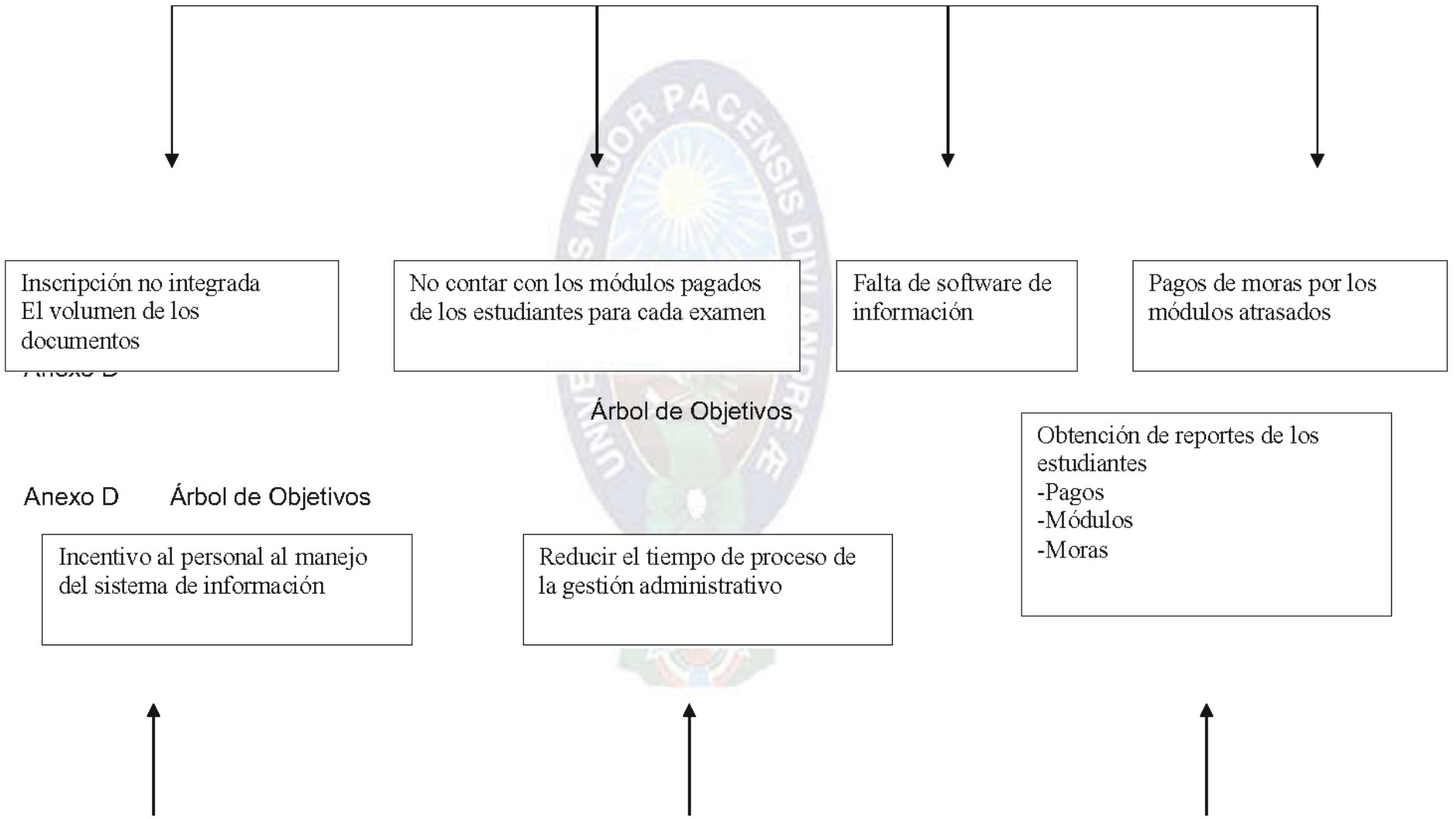
	RESUMEN NARRATIVO	IDENTIFICADORES	MEDIOS DE VERIFICACION	SUPUESTOS
FIN	Mejorar gestión administrativa	Acreditación por la Udabol en 30/03/2008	Acreditación entregado en la decanato de Udabol	La estabilidad del dentro de idiomas ingles de

				la Udabol.
PROPOSITO	Facilitar la transformación de gestión de datos cliente.	Se desarrollara bajo algunos reglamentos de la propia universidad	Manuales de operaciones Mantenimientos lo necesario	Predisposición de las autoridades administrativas. Personal.
COMPONENTES	El control al cliente será siempre necesario y eficiente	Modulo del sistema hasta junio	Documento de especificación de requerimiento entregados a la institución	Software libre
ACTIVIDADES	<ul style="list-style-type: none"> -Informe necesario -Diseño de los módulos del sistema -Implementar y probar el sistema -Manual para el funcionamiento. 	<p>Servidores de la institución 2 PCs. Elaboración pasantía</p>	Adquirirá el centro de idioma ingles.	<p>Financian los mismos estudiantes por los cursos Apoyo de las autoridades para la realización del proyecto.</p>

Anexo C

Árbol de problemas





Anexo D Árbol de Objetivos

```
graph TD; A[Implementación del sistema para el control del cliente del centro de idiomas ingles.] --> B[El sistema permitirá sistematizar los procesos utilizando las herramientas mas adecuadas]; A --> C[El sistema permitirá acceder a la información tanto de los estudiantes como los administrativos]; A --> D[El sistema permitirá la inscripción de alumnos tantos de la misma universidad como los que no son]; A --> E[Permitirá la consulta por la pagina Web.];
```

Implementación del sistema para el control del cliente del centro de idiomas ingles.

El sistema permitirá sistematizar los procesos utilizando las herramientas mas adecuadas

El sistema permitirá acceder a la información tanto de los estudiantes como los administrativos

El sistema permitirá la inscripción de alumnos tantos de la misma universidad como los que no son

Permitirá la consulta por la pagina Web.

Anexo F

Nomina de registro de pagos mensuales de la institución de los estudiantes.

1		REGISTRO DE PAGOS MENSUALES			
		Teacher's: Luis Vasquez		Starter I Turno A de 07:15 a 08:45 (10 de Marzo)	
		DATOS PERSONALES			DATOS DOMICILIO
No.		APELLIDO PATERNO	APELLIDO MATERNO	NOMBRES	DIRECCION
1	1	Palenque	Ramos	Alvaro	
2	1	Alarcon	Aparicio	Marco Andres	
3	1	Prudencio	Rodriguez	Andres	
4	1	Vargas	Mendez	Lenny Eliana	
5	1	Helguero	Ayala	Bryan Adalid	
6	1	mendez	gutierrez	Laura Griselda	
7	1	Caro	Colque	Claudio Marcelo	
8	1	Calpi	Loza	Shirley Priscilla	
9	1	Flores	Valenzuela	Milenka Vanesa	
10	1	Monrroy	montecinos	Jacqueline Fabiola	
11	1	Claros M.		Jade	
12	1	Andrade		Caludia	
13	1	Villalobos		Patricia	
14	1	Gutierrez	Argani	Mayra Lucia	
15	1	Blanco	Sanchez	Karla Veronica	
16	1	Calvo	Omiste	Cecilia Ivonne	
17	1	Lema	Imaña	Caludia Carolina	
18	1	Palacios	Fernandez	Viviana	
19	1	Lopez	Cadena	Reyna Estefani	
20	1	Vera	Cecilia	Cecilia	
	20				

Anexo E

Lista de nomina de estudiantes inscritos

**NOMINDA DE ALUMNOS DE LINGUA
DE TODA LA INSTITUCION.**

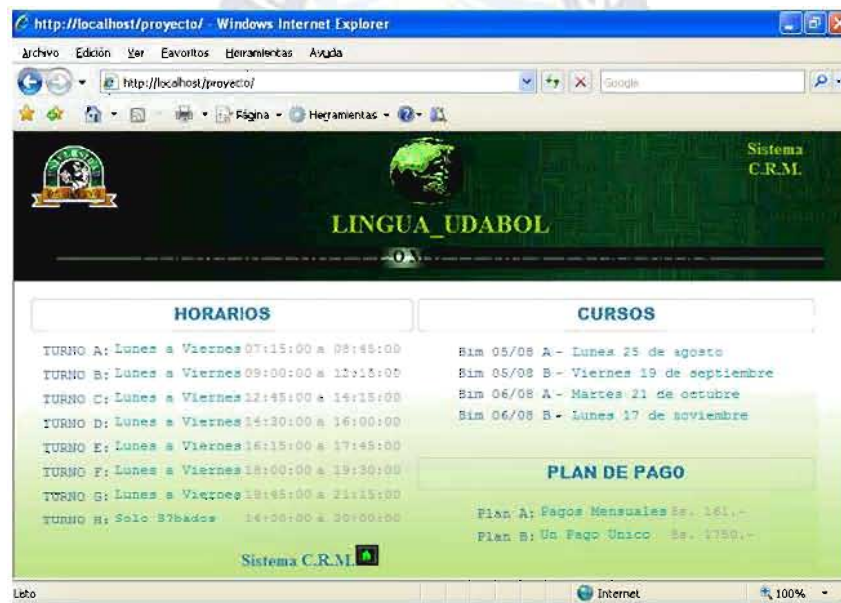
	Apellido Pat	Apellido Mat	Nombre
1	Achu	Cocarico	Orlando
1	Aguilar	Contreras	Jose Pablo
1	Aguilar	Espinoza	Paola Teresa
1	Aguilera	Fernandez	Pool Diego
1	Aguirre	Vargas	Carla Susana
1	Aima	Vertiz Blanco	Claudia Evelin
1	Alanes	Morales	Maria Alejandra
1	Alarcon	Pomier	Roger Igor
1	Alarcon	Saravia	Marcelo Alberto
1	Alba	Moscoso	Betania angela
1	Albarracin	Peralta	Boris Jorge
1	Ali	Mamani	Maria Eugenia
1	Aliaga	Del Carpio	Javier
1	Aliaga	Del Carpio	Adriana
1	Aller	Salcedo	Piedad
1	Altamirano	Cardenas	Elias
1	Alvarez		Iver
1	Alvarez	Alconini	Fabricio
1	Amonzabel	mariscal	Carla
1	Andrade	Calvimontes	Sergio Yamil
1	Apaza	Diaz	Rodrigo Daniel
1	Apaza	Francisco	Oscar Eugenio
1	Apaza	Mejillon	Sara
1	Apaza	mamani	Daysi Virginia
1	Aquino	Paco	Karina
1	Aquize	Quiroz	Erick Rolando
1	Araoz	Perez	Vania Danitza
1	Arcani	Choque	Maria Beatriz
1	Arce	sandoval	mauricio
1	Argandoña	camacho	Ruth
1	Argote	Orozco	Juan Carlos
1	Arias	Sanchez	Brayan Gonzalo
1	Arismendi	Rivera	Edwin

1	Arismendi	solares	Lilian
1	Aruquipa	Calle	Luz Eva
1	Asturizaga	Irusta	Gonzalo Miguel
1	Ayala	Ramirez	Carol milenka
1	Bacarreza	Tejada	Mirko Gilberto
1	Balderrama	Ordoñez	Jannette
1	Ballester	Fernandez	Lionel
1	Ballon	Vasquez	carolina

Anexo G

Pantallas del sistema

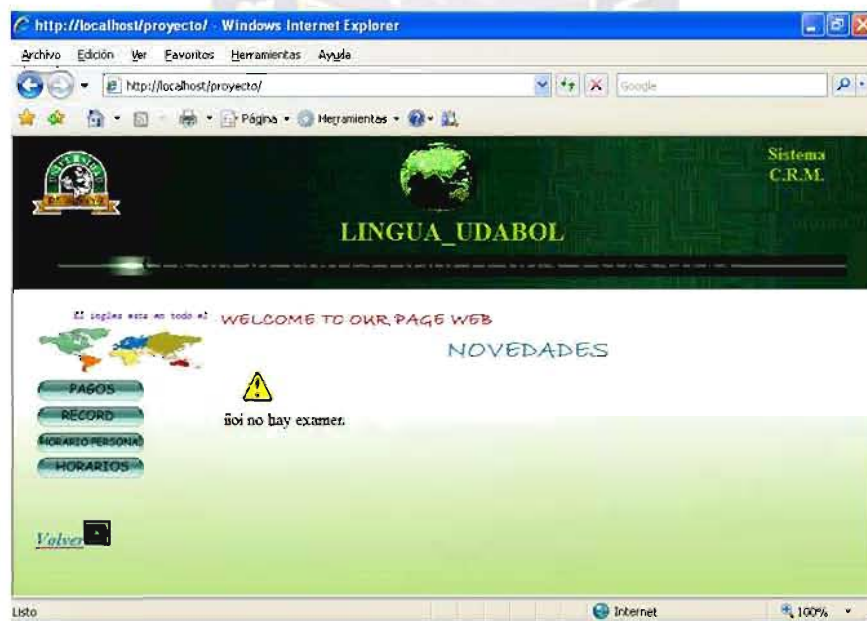
Pantalla principal



Pantallas posteriores de ingreso al sistema



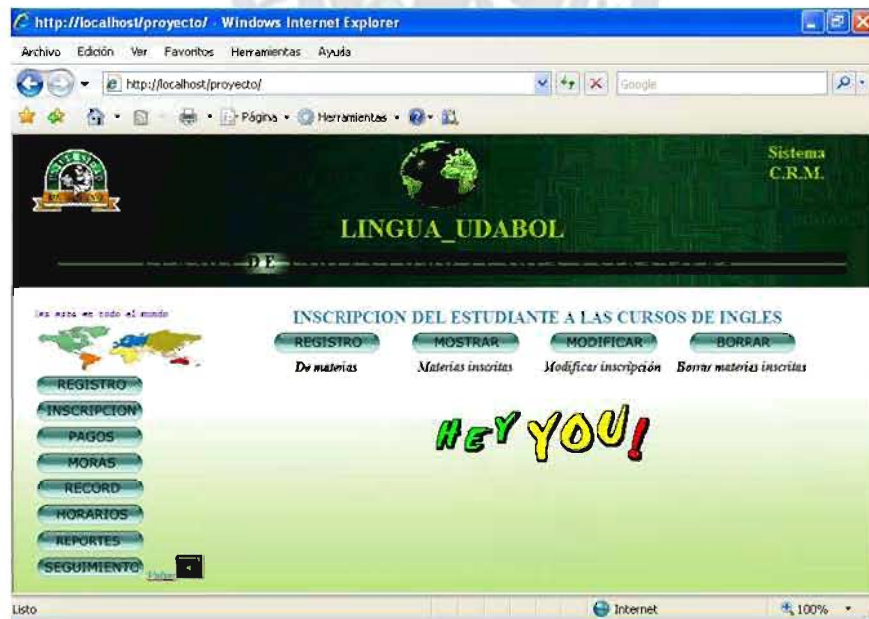
Pantalla de acceso para el estudiante



Pantalla del administrador



Exclusivo para el administrador



http://localhost/proyecto/ - Windows Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

http://localhost/proyecto/ Google

Sistema C.R.M.

LINGUA_UDABOL

REGISTRO INSCRIPCION PAGOS MORAS RECORD HORARIOS REPORTES SEGUIMIENTO

INSCRIPCION DEL ESTUDIANTE A LAS CURSOS DE INGLES

REGISTRO MOSTRAR MODIFICAR BORRAR

De materias Materias inscritas Modificar inscripción Borrar materias inscritas

INSCRIPCION DEL ESTUDIANTE

CI: Estudiantes:

Nombre del Estudiante: Paterno Materno Nombres

Materia a Inscribir: STARTER I

Turno: TURNO A

Fecha de Inscripción: Q1 Enero 2006

Inscribir

Listo Internet 100%

