



**ESCUELA SUPERIOR POLITÉCNICA AGROPECUARIA DE MANABÍ
MANUEL FÉLIX LÓPEZ**

CARRERA INFORMÁTICA

**TESIS PREVIA LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN INFORMÁTICA**

TEMA:

**SISTEMA INFORMÁTICO DE TEST ON-LINE PARA LA COMISIÓN
DE EVALUACIÓN DE LA ESCUELA SUPERIOR POLITÉCNICA
AGROPECUARIA DE MANABÍ MANUEL FÉLIX LÓPEZ**

AUTOR:

ADRIÁN HERÁCLITO ALCÍVAR SANTANDER

TUTORA:

ING. JÉSSICA J. MORALES CARRILLO MG.SC.

CALCETA, FEBRERO 2014

DERECHOS DE AUTORÍA

Adrián Heráclito Alcívar Santander, declaro bajo juramento 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.

A través de la presente declaración cedo mis derechos de propiedad intelectual a la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López, según lo establecido por la Ley de Propiedad Intelectual y su reglamento.

ADRIÁN H. ALCÍVAR SANTANDER

CERTIFICACIÓN DE TUTORA

Jéssica Johanna Morales Carrillo certifica haber tutelado la tesis **SISTEMA INFORMÁTICO DE TEST ON-LINE PARA LA COMISIÓN DE EVALUACIÓN DE LA ESCUELA SUPERIOR POLITÉCNICA AGROPECUARIA DE MANABÍ MANUEL FÉLIX LÓPEZ**, que ha sido desarrollada por Adrián Heráclito Alcívar Santander, previa la obtención del título de Ingeniero en Informática, de acuerdo al **REGLAMENTO PARA LA ELABORACIÓN DE TESIS DE GRADO DE TERCER NIVEL** de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López.

ING. JÉSSICA J. MORALES CARRILLO MG.SC.

APROBACIÓN DEL TRIBUNAL

Los suscritos integrantes del tribunal correspondiente, declaran que han **APROBADO** la tesis **SISTEMA INFORMÁTICO DE TEST ON-LINE PARA LA COMISIÓN DE EVALUACIÓN DE LA ESCUELA SUPERIOR POLITÉCNICA AGROPECUARIA DE MANABÍ MANUEL FÉLIX LÓPEZ**, que ha sido propuesta, desarrollada y sustentada por Adrián Heráclito Alcívar Santander, previa la obtención del título de Ingeniero en Informática, de acuerdo al **REGLAMENTO PARA LA ELABORACIÓN DE TESIS DE GRADO DE TERCER NIVEL** de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López.

ING. ORLANDO AYALA PULLAS
MIEMBRO

ING. EDYS SOLORZANO INTRIAGO
MIEMBRO

ING. DANIEL MERA MARTÍNEZ
PRESIDENTE

AGRADECIMIENTO

A la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Feliz López que me dio la oportunidad de una educación superior de calidad y en la cual e forzado mis conocimientos profesionales día a día;

A mi familia, pilares fundamentales en mi formación académica, humana y personal.

A mis profesores y directivos.

ADRIÁN H. ALCÍVAR SANTANDER

DEDICATORIA

A mis padres, pilares fundamental que con sacrificio diario me permitieron alcanzar una meta más en mi formación académica y personal.

A mis amigos, profesores y compañeros que indirecta o directamente compartieron experiencias y enseñanzas que fueron vitales para la meta alcanzada.

ADRIÁN H. ALCÍVAR SANTANDER

CONTENIDO GENERAL

DERECHOS DE AUTORÍA	ii
CERTIFICACIÓN DE TUTORA.....	iii
APROBACIÓN DEL TRIBUNAL.....	iv
AGRADECIMIENTO.....	v
DEDICATORIA.....	vi
CONTENIDO GENERAL.....	vii
CONTENIDO DE CUADROS Y FIGURAS.....	xi
RESUMEN	xvii
PALABRAS CLAVES	xvii
ABSTRACT	xviii
KEYWORDS	xviii
CAPITULO I. ANTECEDENTES	19
1.1. PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA.....	19
1.2. JUSTIFICACIÓN	21
1.3. OBJETIVOS	24
1.3.1. OBJETIVO GENERAL	24
1.3.2. OBJETIVOS ESPECÍFICOS.....	24
1.4. IDEA A DEFENDER.....	25
2.1. EVALUACIÓN	26
2.1.1. LAS FUNCIONES Y FINES DE LA EVALUACIÓN DEL APRENDIZAJE.....	26
2.1.2. CARACTERÍSTICAS DE LA EVALUACIÓN	27
2.1.3. TIPOS DE EVALUACIÓN	28
2.1.3.1. EVALUACIÓN AL VUELO	28

2.1.3.2.	EVALUACIÓN PLANEADA PARA LA INTERACCIÓN	28
2.1.3.3.	EVALUACIÓN ENCLAVADA EN EL CURRÍCULO	28
2.2.	TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN (TIC)	29
2.2.1.	LA INTEGRACIÓN DE LAS TIC EN LA EDUCACIÓN	29
2.2.2.	FUNCIONES DE LAS TIC EN LA EDUCACIÓN	31
2.3.	SOFTWARE	31
2.3.1.	CLASIFICACIÓN DEL SOFTWARE	32
2.3.1.1.	SOFTWARE DE SISTEMA	32
2.3.1.2.	SOFTWARE DE PROGRAMACIÓN	33
2.3.1.3.	SOFTWARE DE APLICACIÓN	33
2.3.2.	CONCEPTUALIZACIÓN DE LA WEB 2.0	34
2.3.2.1.	WEB 2.0 Y SUS APLICACIONES DIDÁCTICAS	35
2.3.2.2.	E-LEARNING O ELECTRONIC LEARNING	35
2.3.3.	APLICACIÓN WEB	36
2.3.3.1.	TIPOS DE APLICACIONES WEB	37
2.3.4.	LENGUAJES DE PROGRAMACIÓN	37
2.3.5.	LENGUAJES DE PROGRAMACIÓN PARA LA WEB	38
2.3.5.1.	Hypertext Pre-Procesador (PHP)	38
2.3.5.2.	Active Server Pages (ASP)	39
2.3.5.3.	Java Server Pages (JSP)	40
2.3.6.	ENTORNOS DE DESARROLLO INTEGRADO (IDE)	40
2.3.6.1.	NETBEANS	41
2.3.6.2.	CODECHARGE STUDIO	41
2.3.6.3.	SUBLIME TEXT	41
2.3.6.4.	VISUAL STUDIO.NET	42

2.3.7. FRAMEWORK	43
2.3.7.1. ADO .NET ENTITY FRAMEWORK.....	43
2.3.7.2. BOOTSTRAP FRAMEWORK	45
2.3.8. BASES DE DATOS.....	45
2.3.9. SISTEMAS GESTORES DE BASES DE DATOS	46
2.3.9.1. SQL SERVER	47
2.4. INGENIERÍA DE SOFTWARE.....	47
2.4.1. INGENIERÍA DE REQUISITOS	48
2.4.1.1. UML	49
2.4.1.2. DIAGRAMA DE CASOS DE USO.....	50
2.4.2. MODELOS DE PROGRAMACIÓN.....	51
2.4.2.1. PROGRAMACIÓN EN N CAPAS	51
2.4.2.1.1. TIPOS DE CAPAS.....	52
2.4.3. PROGRAMACIÓN ORIENTADA A OBJETOS	53
2.4.4. METODOLOGÍAS DE DESARROLLO AGILES.....	53
2.4.4.1. PROGRAMACIÓN EXTREMA.....	54
2.4.4.1.1. FASES DE LA METODOLOGÍA DE DESARROLLO EXTREME PROGRAMMING XP.....	55
2.4.5. INGENIERÍA WEB	55
2.4.5.1. PROCESOS DE LA IWEB (INGENIERÍA WEB)	57
2.4.5.2. MÉTODOS DE LA IWEB	58
CAPITULO III. DESARROLLO METODOLÓGICO	60
3.1. METODOLOGÍA DE INVESTIGACIÓN.....	60
3.1.1. MÉTODO INDUCTIVO.....	60
3.1.2. MÉTODO DEDUCTIVO	61

3.2. METODOLOGÍA INFORMÁTICA	61
3.2.1. FASE DE PLANEACIÓN.....	61
3.2.1.1. ESTUDIO DE FACTIBILIDAD.....	61
3.2.1.2. OBTENCIÓN Y ANÁLISIS DE REQUERIMIENTOS.....	62
3.2.1.3. VALIDACIÓN DE REQUERIMIENTOS.....	63
3.2.2. FASE DE DISEÑO	83
3.2.3. FASE DE CODIFICACIÓN.....	86
3.2.4. FASE DE PRUEBA E IMPLEMENTACIÓN.....	92
CAPÍTULO IV. RESULTADOS Y DISCUSIÓN.....	94
DISCUSIÓN	111
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES	112
BIBLIOGRAFÍA	114
ANEXOS	119

CONTENIDO DE CUADROS Y FIGURAS

Figura 2.1. Arquitectura ado.net entity framework.....	42
Figura 2.2. Elementos del daigrama de caso de uso.....	48
Figura 3.1. Identificación de actores del sistema.....	60
Figura 3.2. Diagrama general de fnciones del sistema.....	61
Figura 3.3. Diagrama de validación de usuario.....	62
Cuadro 3.1. Casos de uso validación de usuario, Inicio sesión.....	62
Cuadro 3.2 Casos de uso validación de usuario, Cerrar sesión.....	63
Figura 3.2. Caso de uso. Gestión de Sub Áreas.....	64
Cuadro 3.3. Caso de uso Gestión de Sub Áreas.....	65
Figura 3.5. Diagrama de Gestión de Temas.....	66
Cuadro 3.4. Caso de uso Gestión de Temas.....	66
Figura 3.6. Diagrama de gestión de preguntas.....	67
Cuadro 3.5. Caso de uso Gestion de Preguntas.....	68

Figura	3.7.	Diagrama	de	gestión	de	
Test.....						69
Cuadro	3.6.	Caso	de	uso,	Gestion	de
Test.....						69
Cuadro 3.7. Caso de uso, Asignacion de preguntas.....						70
Figura	3.8.	Diagrama	de	toma	de	
Test.....						71
Cuadro	3.8.	Caso	de	uso,	Tomar	
Test.....						71
Cuadro	3.9.	Caso	de	uso,	Visualizacion	de
resultados.....						72
Figura	3.9	Diagrama	de	gestión	de	
Reporte.....						73
Cuadro	3.10	Caso	de	uso,	Gestionar	
Reportes.....						73
Cuadro	3.11.	Diccionario	de	datos	de	la
“Alumno”.....						tabla
						74
Cuadro	3.12.	Diccionario	de	datos	de	la
“Asignatura”.....						tabla
						75
Cuadro	3.13.	Diccionario	de	datos	de	la
“Carrera”.....						tabla
						75
Cuadro	3.14.	Diccionario	de	datos	de	la
“Distribución”.....						tabla
						75
Cuadro	3.15.	Diccionario	de	datos	de	la
“Docente”.....						tabla
						75

Cuadro	3.16.	Diccionario	de	datos	de	la	tabla	
“Instrumento”								75
Cuadro	3.17.	Diccionario	de	datos	de	la	tabla	
“Matricula”								76
Cuadro	3.18.	Diccionario	de	datos	de	la	tabla	“Periodo
Lectivo”								76
Cuadro	3.19.	Diccionario	de	datos	de	la	tabla	
“Persona”								76
Cuadro	3.20.	Diccionario	de	datos	de	la	tabla	
“Preguntas”								77
Cuadro	3.21.	Diccionario	de	datos	de	la	tabla	intermedia
“Preguntas_Test”								77
Cuadro	3.22.	Diccionario	de	datos	de	la	tabla	
“Prueba”								77
Cuadro	3.23.	Diccionario	de	datos	de	la	tabla	
“PreguntaTest”								78
Cuadro	3.24.	Diccionario	de	datos	de	la	tabla	
“Rangos”								78
Cuadro	3.25.	Diccionario	de	datos	de	la	tabla	
“Semestre”								78
Cuadro	3.26.	Diccionario	de	datos	de	la	tabla	intermedia
“Semestre_Test”								78
Cuadro	3.27.	Diccionario	de	datos	de	la	tabla	“Sub
Área”								78

Cuadro 3.28. Diccionario de datos de la tabla intermedia “SubArea_Prueba”.....	78
Cuadro 3.29. Diccionario de datos de la tabla “Tema”.....	79
Cuadro 3.30. Diccionario de datos de la tabla “Temas Prueba”.....	79
Cuadro 3.31. Diccionario de datos de la tabla “Tipo”.....	79
Cuadro 3.32. Diccionario de datos de la tabla “Tipo Calificación”.....	79
Cuadro 3.33. Diccionario de datos de la tabla “Tipo Test”.....	79
Figura 3.10 Consulta en SQL Server para obtener estudiantes con semestre.....	80
Figura 3.11 Consulta en SQL Server para obtener resultados de evaluación.....	80
Cuadro 3.34. Guion técnico del módulo Ingreso al sistema.....	81
Cuadro 3.35. Guion técnico del menú de administración.....	82
Figura 3.12. Plantilla base del sistema.....	83
Figura 3.13. Captura de pantalla que muestra las tres capas del sistema.....	84
Figura 3.14. Modelo de la base de datos generado mediante Entity Framework.....	84

Figura 4.7. Diagrama de clases tontas de la capa negocio.....	98
Figura 4.8. Diagrama de clases de catálogos de la capa negocio.....	99
Figura 4.9. Diagrama de clases de la capa de datos.....	100
Figura 4.10. Pantalla web de evaluación.....	101
Figura 4.11. Tabla resultante al terminar la evaluación.....	101
Foto 4.12. Resultados evidenciados segundos después de terminar de evaluar...	102
Grafico 4.1. Grafico comparativo de tiempos de diseño, evaluación y calificación de un test	102
Figura 4.13. Prueba de rendimiento del sistema.....	104
Figura 4.14. Prueba de rendimiento del sistema.....	104
Cuadro 4.1. Datos relacionados al análisis costo beneficio.....	105
Grafico 4.2. Grafico comparativo costo real, esperado.....	106

RESUMEN

Con el objetivo de desarrollar un sistema informático de evaluaciones en línea para la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López, se realizó un análisis de métodos de evaluación, utilizados por el comité de evaluación institucional para la valoración interna de los estudiantes. Para la elaboración del sistema y demostrar que los resultados están respaldados técnicamente se trabajó con la metodología de desarrollo Proceso XP (Programación extrema) misma que consta de cuatro fases, dentro de las cuales se realizó desde la toma de requerimientos, con la colaboración continua del comité de evaluación donde se analizaron y clasificaron los requerimientos y consideraciones que el software debe cumplir , terminada esta fase se realizó el diseño de un prototipo con la finalidad de agilizar y garantizar que la interfaz de usuario facilite la toma de evaluaciones, posteriormente se codificaron y validaron los componentes utilizados en el sistema haciendo uso de Bootstrap para el entorno visual y modo de interacción con el usuario, Entyti Framework, C#, JavaScript, Ajax y el paradigma de la programación orientada a objetos para darles funcionalidad respectivamente, validándolos mediante HTML5, JavaScript y clases respectivamente, y de esta manera perfeccionar el desarrollo con la fase de pruebas, que permitió evidenciar la reducción de un 95% en los tiempos de crear, configurar y asignar reactivos a una evaluación, y en 99% el proceso de calificación y emisión reportes de resultados.

PALABRAS CLAVES

Evaluación, Software de evaluación, Evaluación interna, Generador de test, Test ON-LINE.

ABSTRACT

In order to develop the computer system online assessments for the Evaluation Commission of Agricultural Polytechnic School of Manabí Manuel Felix Lopez, an analysis of assessment methods used by the accreditation committee for internal valuation was performed students. For efficient processing of the system and show that the results are supported technically worked with the methodology development process XP (Extreme Programming) which consists of four phases , within which was held from requirements gathering , with continuous collaboration evaluation committee which analyzed and classified the requirements and considerations that software must meet , completed this phase the design and adoption of a prototype was made in order to expedite and ensure that the user interface facilitates decision- evaluations later were programmed and validated components used in the system using Bootstrap to the visual environment and mode of interaction with the user, Entyti Framework, C # , JavaScript, Ajax and the paradigm of object-oriented to give functionality respectively programming , validating them by HTML5 , JavaScript and classes respectively, and thus enhance the development with the testing phase that ensured compliance with guidelines and characteristics considered in the initial development , the evidence allowed a 95% reduction in time to create , configure, and assign reagents to an evaluation , and 99% in the rating process and reports results.

KEYWORDS

Assessment, Evaluation Software, Internal Assessment, Test Generator, Test ON-LINE.

CAPITULO I. ANTECEDENTES

1.1. PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA

Durante las últimas décadas, varios factores hicieron que las universidades tuvieran que preocuparse mucho más de la calidad de sus actividades. La causa principal parece ser la necesidad de fortalecer las instituciones y los sistemas de enseñanza superior a raíz de la rápida expansión de las últimas décadas; relacionadas con el papel más importante que desempeñan los establecimientos de enseñanza superior en el desarrollo de una economía competitiva (L'écuyer, 2001).

La Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López en su corta vida institucional se ha caracterizado por el excelente nivel académico de sus profesores y estudiantes, pero; las nuevas leyes aprobadas en materia de educación en el estado ecuatoriano en los últimos años, han generado un proceso de continua superación y evaluación tanto interna como externa, y como lo menciona Rubio, *et al.*, (2013) la cuestión general de la evaluación y de la ayuda al aprendizaje se ha ido situando como algo central en muchas visiones de la docencia universitaria, especialmente en aquellas que buscan su renovación. Estos procesos necesitan de datos que le permita al comité de evaluación institucional que es el órgano encargado de organizar las actividades orientadas a la acreditación institucional, tener la información necesaria respecto al rendimiento de los y las estudiantes, y en base a esta indagación tomar decisiones en las áreas donde se detecten debilidades y así aprobar las constantes evaluaciones que la reconozca como universidad acreditada del Sistema de Educación Superior del Ecuador.

Las tecnologías de información y comunicación ya no son actores pasivos del sistema de educación, por el contrario día a día abarcan más áreas dentro del transcurso de enseñanza y aprendizaje, en este campo en el comité de evaluación de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López se evidencia la necesidad de una herramienta tecnológica que les facilite el proceso de una evaluación, y con los datos resultantes les permita conocer la adquisición de las capacidades y habilidades dadas en clases, con el afán de detectar y tomar los correctivos en las áreas donde se muestren deficiencias.

Con los antecedentes mencionados el autor se formula la siguiente interrogante.

¿De qué manera mejorar el proceso de evaluación de los estudiantes de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López?

1.2. JUSTIFICACIÓN

García (2011) expresa que en los contextos educativos tanto en el ámbito formal como en los procesos de la información de carácter no formal, las TIC pueden ayudar a los estudiantes a adquirir una serie de capacidades y competencias.

El autor citando la Constitución de la República del Ecuador en su artículo 346 dicta que existirá una institución pública, con autonomía, de evaluación integral, interna y externa, que promueva la calidad de la educación.

Que, la Constitución de la República del Ecuador en su artículo 347, literal 8 expresa que será responsabilidad del Estado Incorporar las tecnologías de la información y comunicación en el proceso educativo.

Que, la Constitución de la República del Ecuador en su artículo Art. 353 literal 2 pronuncia que el sistema de educación superior se regirá por un organismo público técnico de acreditación y aseguramiento de la calidad de instituciones, carreras y programas, que no podrá conformarse por representantes de las instituciones objeto de regulación.

Que, la Ley Orgánica de Educación Superior en su Art. 13 sobre las funciones del Sistema de Educación Superior establece: Evaluar, acreditar y categorizar a las instituciones del Sistema de Educación Superior, sus programas y carreras, y garantizar independencia y ética en el proceso.

Que, la Ley Orgánica de Educación Superior en su Art. 14 literal a define que son instituciones del Sistema de Educación Superior las universidades, escuelas politécnicas públicas y particulares, debidamente evaluadas y acreditadas, conforme la presente Ley.

Que, la ley Orgánica de Educación Superior en su Art. 99 menciona que la Autoevaluación es el riguroso proceso de análisis que una institución realiza sobre la totalidad de sus actividades institucionales o de una carrera, programa o posgrado específico, con amplia participación de sus integrantes, a través de un análisis crítico y un diálogo reflexivo, a fin de superar los obstáculos existentes y considerar los logros alcanzados, para mejorar la eficiencia institucional y mejorar la calidad académica.

Que, la ley Orgánica de Educación Superior en su Art. 100 dicta que la evaluación externa es el proceso de verificación que el Consejo de Evaluación, Acreditación y Aseguramiento de la Calidad de la Educación Superior realiza a través de pares académicos de la totalidad o de las actividades institucionales o de una carrera o programa para determinar que su desempeño cumple con las características y estándares de calidad de las instituciones de educación superior y que sus actividades se realizan en concordancia con la misión, visión, propósitos y objetivos institucionales o de carrera, de tal manera que pueda certificar ante la sociedad la calidad académica y la integridad institucional.

Que, la ley Orgánica de Educación Superior en su Art. 103 determina el examen Nacional de evaluación de carreras y programas académicos.- Para efectos de evaluación se deberá establecer un examen para estudiantes de último año, de los programas o carreras. El examen será complementario a otros mecanismos de evaluación y medición de la calidad.

Que, la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López en su artículo 2 del reglamento de tesis de grado enuncia que: Todo tema de tesis de grado estará relacionado con las líneas de investigación de la carrera del postulante, enmarcado en las áreas y prioridades de investigación establecidas por la ESPAM MFL en concordancia con el Plan Nacional para el Buen Vivir.

Ante lo expuesto, el autor consideró que en la actualidad las TIC están cambiando la manera de comunicarse entre las personas y de la mano con este vertiginoso avance, el comité de evaluación institucional de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López, ve la necesidad de utilizar equipos y herramientas tecnológicas que optimicen el constante proceso de evaluación que las Universidades y Escuelas Politécnicas del sistema de educación superior deben cumplir.

Se creó una aplicación que permitió generar test y rendir evaluaciones on-line, de tal forma que el comité de evaluación institucional creó y diseñó test digitalmente, luego evaluó y evidenció los resultados de los conocimientos de los estudiantes utilizando las TIC, este proceso ayudó a reducir de manera significativa el tiempo que le tomaba calificar manualmente las pruebas y exámenes que se aplican a los estudiantes, es así que se hizo indispensable y necesaria la creación del sistema que cambió el viejo modelo de evaluar a los estudiantes, que consistía desde la elaboración del banco de preguntas, selección de las que se tomarán en el examen, el diseño del formulario de preguntas con su respectiva hoja de respuestas, copias, gasto de recursos materiales, físicos y otros como tiempo; por un modelo permitió escoger y diseñar el test digitalmente desde un banco de preguntas sin la necesidad de invertir mayor tiempo y con la seguridad de elementos de validación que permitieron a los actores del proceso confiar en los resultados del mismo.

El modelo de toma de test se encontró enfocado en un constante avance y compromiso de cuidar y respetar la naturaleza, esto se evidenció ya que al rendir los exámenes digitalmente se ahorraron costos de impresiones, tiempo en el proceso de diseño del test y en el proceso de calificación, con ello se fomentó la cultura “cero papel”, una tendencia creciente en la actualidad por la indiscriminada tala de árboles.

1.3. OBJETIVOS

1.3.1. OBJETIVO GENERAL

Desarrollar un sistema de Test On-Line para la Comisión de Evaluación de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López en el mejoramiento del proceso de valoración de los estudiantes.

1.3.2. OBJETIVOS ESPECÍFICOS

- Realizar el análisis de requerimientos del sistema gestor de test y establecer las funcionalidades del mismo.
- Crear la aplicación que permita diseñar cuestionarios interactivos y el módulo para su rendimiento web.
- Implementar el sistema.
- Probar la funcionalidad y validar el sistema.

1.4. IDEA A DEFENDER

La aplicación de test on-line mejora el proceso de evaluación de los estudiantes de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López.

CAPÍTULO II. MARCO TEÓRICO

2.1. EVALUACIÓN

No sería demasiado arriesgado afirmar, que uno de los elementos que están caracterizando la educación en todo el mundo, en estos primeros años del siglo XXI, es el énfasis puesto en los sistemas de evaluación de la calidad. De esta forma, prácticamente en la totalidad de los países de América Latina existen centros, unidades, institutos o departamentos ligados a los ministerios de educación, destinados a evaluar la calidad de la educación de sus sistemas. Es así que las evaluaciones internacionales, nacionales o regionales se multiplican constantemente (Murillo y Román, 2010).

Así mismo, Tardif (2006) manifiesta que la evaluación en la formación por competencias pasa por considerar que el objeto de la evaluación no son sólo los conocimientos adquiridos sino también, y hasta diría que sobre todo, las competencias desarrolladas por los estudiantes, tiene una función reguladora del aprendizaje, no es una simple actividad técnica, sino que constituye un elemento clave en la calidad de los aprendizajes.

Considerando los conceptos expuestos anteriormente, se puede definir a la evaluación como un elemento clave en el proceso enseñanza aprendizaje, por el volumen de información que facilita al profesor y por las consecuencias que tiene para el docente, el alumnado y el sistema educativo en que están integrados.

2.1.1. LAS FUNCIONES Y FINES DE LA EVALUACIÓN DEL APRENDIZAJE

El objetivo de la evaluación del aprendizaje, como actividad genérica, es valorar el aprendizaje en cuanto a sus resultados y consecución. Las finalidades o fines

marcan los propósitos que signan esa evaluación. Las funciones están referidas al papel que desempeña para la sociedad, para la institución, para el proceso de enseñanza aprendizaje, para los individuos implicados en el mismo (Murillo y Román, 2010).

La evaluación formativa es un proceso sistemático para obtener evidencia continua acerca del aprendizaje. Los datos son usados para identificar el nivel de aprendizaje real del alumno y adaptar la clase para ayudarlo a alcanzar las metas de aprendizaje deseadas. En la evaluación formativa, los alumnos son participantes activos con sus profesores, comparten metas de aprendizaje y entienden cómo progresan, cuáles son los siguientes pasos que necesitan dar y cómo darlos (Moreno, 2011).

2.1.2. CARACTERÍSTICAS DE LA EVALUACIÓN

Tardif (2006) propone ver la evaluación de competencias como un planteamiento videográfico y no fotográfico. En efecto, no se trata tanto de emitir un juicio al final del trayecto como de seguir la progresión del desarrollo de competencias. En este sentido, la evaluación formativa, que informa al estudiante sobre la progresión de su aprendizaje, es un elemento esencial de todo dispositivo de evaluación en una formación por competencias. La evaluación de competencias se basa entonces en el acceso a fuentes múltiples y variadas de información con el fin de determinar si los estudiantes han alcanzado el nivel esperado de desarrollo de competencias, así como un grado suficiente de dominio de los recursos vinculados a cada competencia. Como la competencia es un “saber actuar complejo en situación”, la evaluación de las competencias deberá realizarse en una situación lo más auténtica posible. Por situación auténtica, nos referimos a una situación lo más cercana posible al contexto profesional en el que los estudiantes podrán evolucionar una vez que hayan finalizado sus estudios. Estas situaciones

auténticas de evaluación permiten a los estudiantes comprometerse en la realización de una tarea completa compleja y significativa (Fernández, 2010).

2.1.3. TIPOS DE EVALUACIÓN

De acuerdo con Heritage (2007) citado por Moreno (2011), la evaluación formativa incluye una variedad de estrategias para obtener evidencia, la cual puede ser categorizada en tres amplios tipos: evaluación al vuelo, evaluación planeada para la interacción, y evaluación enclavada en el currículum.

2.1.3.1. EVALUACIÓN AL VUELO

Esta evaluación ocurre espontáneamente durante una clase. Por ejemplo, una profesora escucha las discusiones de un grupo, oye a los alumnos expresar sus ideas equivocadas acerca de un concepto científico que ha estado enseñando entonces, cambia la dirección de su clase para dar una explicación rápida e "inesperada" le permite a la profesora aclarar las ideas equivocadas antes de continuar con su secuencia de clase prevista (Moreno, 2011).

2.1.3.2. EVALUACIÓN PLANEADA PARA LA INTERACCIÓN

En este tipo de evaluación los profesores deciden de antemano cómo aclarar las ideas de los alumnos durante la enseñanza. Por ejemplo, los profesores planifican las preguntas que harán durante la clase a fin de capacitar a los alumnos para explorar ideas, y estas ideas pueden aportar información valiosa para la evaluación (Moreno, 2011).

2.1.3.3. EVALUACIÓN ENCLAVADA EN EL CURRÍCULO

Hay dos tipos de evaluaciones insertas en el currículo, aquellas que los diseñadores del currículo establecieron para solicitar retroalimentación en los puntos clave en una secuencia de aprendizaje, y aquellas que son parte de las actividades continuas de aula. Por ejemplo, las representaciones matemáticas de un alumno creadas durante las clases pueden funcionar como evaluaciones formativas, como también pueden serlo los cuadernos de ciencias, que son parte de las actividades de aula regulares de los alumnos (Moreno, 2011).

2.2. TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN (TIC).

Las TIC son el conjunto de tecnologías que permiten la adquisición, producción, almacenamiento, tratamiento, comunicación, registro y presentación de informaciones, en forma de voz, imágenes y datos contenidos en señales de naturaleza acústica, óptica o electromagnética (Valiente y Gonzáles, 2013).

El recurso que hoy se considera más valioso es la información. Si se quiere alcanzar un objetivo, es preciso acceder a la información pertinente para llegar a tomar las decisiones adecuadas. Puede decirse, que la Sociedad de la Información es, ante todo, Sociedad de formación por ello hoy las TIC pueden ser consideradas esencialmente como el substrato hará la formación de los individuos en esta sociedad. A su vez esta sociedad se va formando moldeada por las TIC (Pérez y Fernández, 2005).

2.2.1. LA INTEGRACIÓN DE LAS TIC EN LA EDUCACIÓN

El uso de las tecnologías de la información y la comunicación en las aulas, se ha ido implementando paulatinamente y en la actualidad constituyen herramientas con habitual presencia en la enseñanza, están transformando la vida personal y profesional, están cambiando las formas de acceso al conocimiento y de

aprendizaje, los modos de comunicación y la manera de relacionarnos, a tal punto que la generación, procesamiento y transmisión de información se está convirtiendo en factor de poder y productividad en la "sociedad informacional". La productividad y la competitividad dependen cada vez más de la capacidad de generar y aplicar la información basada en el conocimiento.

La concepción de aprendizaje y, en consecuencia, las estrategias de enseñanza dirigidas a promoverlo, también han experimentado cambios sustanciales. Mientras el papel de la enseñanza tradicional era proporcionar información, la función del alumno se restringía a asimilarla mediante la práctica y la repetición, fuera de su contexto real de utilización. Hoy, por el contrario, se considera que el aprendizaje no puede ser transmitido sino que debe ser construido por el propio individuo (constructivismo). Por ello, los sistemas innovadores de enseñanza enfatizan el aprendizaje basado en la actividad significativa, entre estos sistemas se encuentran: la enseñanza basada en el aprendizaje de oficios, el aprendizaje basado en problemas; escenarios dirigidos a un objetivo; estructura anclada. (Escontrela y Stojanovic, 2004).

Tal como lo menciona Katz (2009) el impacto de las TIC en educación cubre numerosas áreas de aplicación desde la educación a distancia a la utilización de terminales portátiles y herramientas informáticas en las escuelas. En el caso de la educación a distancia, la investigación muestra, en primer lugar, que alumnos en áreas remotas cuyo vehículo primario de instrucciones de la educación a distancia tienden a desempeñarse a aquellos que se benefician del medio presencial (instrucciones en el aula). Los estudios de Witherspoon *et al.*, (1993) muestran que, pese a la distancia y la separación física con el maestro, los alumnos del sistema a distancia tienden a estar más motivados para el aprendizaje, son más maduros mientras que el diseño del material educativo tiende a ser más sistemático y orientado a enfatizar el proceso de enseñanza.

2.2.2. FUNCIONES DE LAS TIC EN LA EDUCACIÓN

Las TIC ofrecen grandes posibilidades al mundo de la Educación. Pueden facilitar el aprendizaje de conceptos y materias, pueden ayudar a resolver problemas y pueden contribuir a desarrollar las habilidades cognitivas. Las áreas de aplicación de todas estas técnicas es lo que normalmente se denomina Informática Educativa (Pérez, 2005).

La “sociedad de la información” en general y las nuevas tecnologías en particular inciden de manera significativa en todos los niveles del mundo educativo. Las nuevas generaciones van asimilando de manera natural esta nueva cultura que se va conformando y que para nosotros conlleva muchas veces importantes esfuerzos de formación, de adaptación y de “desaparecer” muchas cosas que ahora “se hacen de otra forma” o que simplemente ya no sirven. Los más jóvenes no tienen el peso experiencial de haber vivido en una sociedad “más estática”, de manera que para ellos el cambio y el aprendizaje continuo para conocer las novedades que van surgiendo cada día es lo normal. Precisamente para favorecer este proceso la educación y el proceso enseñanza aprendizaje debe integrar también la nueva cultura digital (Escontrelas y Stojanovic, 2004).

2.3. SOFTWARE

Existen varias definiciones similares aceptadas para software, pero probablemente la más formal sea la siguiente: Es el conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de computación. Considerando esta definición, el concepto de software va más allá de los programas de computación en sus distintos estados: código fuente, binario o ejecutable; también su documentación, los datos a procesar e incluso la información de usuario forman parte del software:

es decir, abarca todo lo intangible, todo lo «no físico» relacionado (Pressman, 2010).

Y como menciona Santos, *et al.* (2009) existe una importante línea de desarrollo de software en y para la web en las que se incluyen las llamadas aplicaciones web, en esencia aplicaciones tradicionales de un PC pero implementadas con la tecnología web y que únicamente requieren un navegador estándar para su utilización. Seguramente el ejemplo más representativo es Google Docs (www.docs.google.com) que ofrece hoja de cálculo, procesador de texto, presentaciones y formularios de forma gratuita, y que junto con otras soluciones de la empresa conforman una interesante oferta para las instituciones académicas.

2.3.1. CLASIFICACIÓN DEL SOFTWARE

Si bien esta distinción es, en cierto modo, arbitraria, y a veces confusa, a los fines prácticos se puede clasificar al software en tres grandes tipos (Pressman, 2010).

2.3.1.1. SOFTWARE DE SISTEMA

Su objetivo es desvincular adecuadamente al usuario y al programador de los detalles de la computadora en particular que se use, aislándolo especialmente del procesamiento referido a las características internas de: memoria, discos, puertos y dispositivos de comunicaciones, impresoras, pantallas, teclados, etc. El software de sistema le procura al usuario y programador adecuadas interfaces de alto nivel, herramientas y utilidades de apoyo que permiten su mantenimiento. Incluye entre otros:

- Sistemas operativos
- Controladores de dispositivos
- Herramientas de Corrección y Optimización

- Servidores
- Utilidades

2.3.1.2. SOFTWARE DE PROGRAMACIÓN

Es el conjunto de herramientas que permiten al programador desarrollar programas informáticos, usando diferentes alternativas y lenguajes de programación, de una manera práctica. Incluye entre otros:

- Editores de texto
- Compiladores Intérpretes
- Enlazadores
- Depuradores
- Entornos de Desarrollo Integrados (IDE): Agrupan las anteriores herramientas, usualmente en un entorno visual, de forma tal que el programador no necesite introducir múltiples comandos para compilar, interpretar, depurar, etc. Habitualmente cuentan con una avanzada interfaz gráfica de usuario (GUI).

2.3.1.3. SOFTWARE DE APLICACIÓN

Es aquel que permite a los usuarios llevar a cabo una o varias tareas específicas, en cualquier campo de actividad susceptible de ser automatizado o asistido, con especial énfasis en los negocios. Incluye entre otros:

- Aplicaciones para Control de sistemas y automatización industria
- Aplicaciones ofimáticas
- Software educativo
- Software empresarial
- Bases de datos

- Telecomunicaciones (por ejemplo Internet y toda su estructura lógica)
- Videojuegos
- Software médico
- Software de Cálculo Numérico y simbólico.
- Software de Diseño Asistido (CAD)
- Software de Control Numérico (CAM) (Pressman, 2010).

2.3.2. CONCEPTUALIZACIÓN DE LA WEB 2.0

En la actualidad la web se ha convertido en una red de computadoras a nivel mundial a través de la cual se puede obtener información sobre cualquier tema y gestionar cualquier tipo de proceso de la vida diaria que represente un problema al realizarlo de forma presencial, ya sea efectuar compras en el mercado, reservar un vuelo, realizar un pedido a un restaurant, etc. (Ponce y Fernández, 2012).

Con el término Web 2.0, subrayamos un cambio de paradigma sobre la concepción de Internet y sus funcionalidades, que ahora abandonan su marcada unidireccionalidad y se orientan más a facilitar la máxima interacción entre los usuarios y el desarrollo de redes sociales (tecnologías sociales) donde puedan expresarse y opinar, buscar y recibir información de interés, colaborar y crear conocimiento (conocimiento social), compartir contenidos. Podemos distinguir:

- Aplicaciones para expresarse/crear y publicar/difundir: blog, wiki.
- Aplicaciones para publicar/difundir y buscar información: podcast, YouTube, Flickr, SlideShare, Del.icio.us.
- Aplicaciones para buscar/acceder a información de la que nos interesa estar siempre bien actualizados: RSS, XML, Atom, Bloglines, GoogleReader, buscadores especializados, Redes sociales, BSCW, Ning, Second Life, Twitter.

- Otras aplicaciones son: line Web 2.0: Calendarios, geolocalización, libros virtuales compartidos, noticias, ofimática online, etc. (Marqués, 2007).

2.3.2.1. WEB 2.0 Y SUS APLICACIONES DIDÁCTICAS

De la Torre (2006) expresa que paralelamente al comienzo de la incorporación de las TIC a las prácticas educativas, se ha debatido bastante sobre las competencias tecnológicas que los docentes debían adquirir en sus diferentes procesos formativos. Sobre todo por ser necesarias determinadas destrezas en el uso y, sobre todo, en la generación de recursos para la Web.

Nadie duda que Internet y la web sea uno de los fenómenos informativos y culturales más espectaculares de los últimos tiempos. De tanto hablar de ellos los hemos convertido en un objeto poliédrico, en una realidad con tantas caras como perspectivas se utilizan para definirlo o aproximarse a él, internet y más concretamente la Web constituye actualmente el sistema más utilizado para la distribución de información en línea. Por otro lado, no ofrece tan solo información, sino que tiene especialmente desarrollados los servicios de comunicación y transacciones entre usuarios u organizaciones, con lo cual se trata del sistema de distribución más completo (Abadal, 2001).

2.3.2.2. E-LEARNING O ELECTRONIC LEARNING

Tipificando e-learning como objeto de un observatorio de estudio no es una idea descontextualizada, sino que se basa, implícitamente y explícitamente, en una serie de hipótesis que pueden ser tipificadas, los entornos de e-learning añaden cambios a la educación y determina nuevos enfoques de la atención. Para que el aprendizaje implique una serie de nuevos objetivos, procesos, condensas y

condiciones, así como el surgimiento de nuevos actores sociales y los nuevos enfoques de los roles tradicionales de educación (Suárez, 2010).

Según Cabero (2006) expresa que el e-Learning se nos presenta como una de las estrategias formativas que puede resolver muchos de los problemas educativos con que nos encontramos, que van desde el aislamiento geográfico del estudiante de los centros del saber hasta la necesidad de perfeccionamiento constante que nos introduce la sociedad del conocimiento.

Céspedes y González (2012) mencionan que este término se refiere a la transferencia de experiencias de aprendizaje a través del uso de nuevas tecnologías de la información y la comunicación. Una de esas tecnologías es el Internet y la utilización de plataformas de aprendizaje, pero también podría incluirse las tecnologías Multimedia o los Simuladores. Una de las mayores ventajas del e-learning es la facilidad de acceso, es decir, cualquier persona puede aprender desde cualquier lugar y en cualquier momento. Esta nueva modalidad de aprendizaje se da a través de la Web combinando herramientas síncronas y asíncronas de comunicación, de ninguna manera esta nueva modalidad va a reemplazar a la modalidad tradicional, sin embargo en la actualidad se la utiliza como complemento o soporte a la modalidad presencial. El e-learning, está definido por tres ejes fundamentales que son: la comunicación, el conocimiento y la tecnología. Esta nueva forma de aprender establece una interacción continua entre el alumno y el profesor cambiando totalmente sus roles de la modalidad presencial.

2.3.3. APLICACIÓN WEB

Oliveros *et al.*, (2011) expresa que, una aplicación web es un sistema de software al que se accede a través de Internet (o Intranet): las aplicaciones Web

constituyen una clase especial de aplicaciones de software que se construyen de acuerdo con ciertas tecnologías y estándares.

Estas aplicaciones pueden utilizarse en la enseñanza de muy diversas formas, por ejemplo, como soporte de presentaciones presenciales o a distancia de profesores o alumnos, como soporte a grupos en el desarrollo de proyectos, facilitando las actividades de supervisión y seguimiento por parte del profesor de los trabajos de los alumnos, como herramientas de test o cuestionarios online, etc. (Santos *et al.*, 2009).

2.3.3.1. TIPOS DE APLICACIONES WEB

De acuerdo a lo que expresa Oliveros *et all.* (2011) una tipología bastante difundida identifica los siguientes tipos de aplicaciones Web:

- Sitios Web centrados en documentos
- Aplicaciones Web Interactivas
- Aplicaciones Web transaccionales
- Aplicaciones Web basadas en flujos de trabajos
- Aplicaciones Web de colaboración
- Web social.
- Aplicaciones Web orientadas a portales.
- Aplicaciones de Web ubicua.
- Web semántica.

2.3.4. LENGUAJES DE PROGRAMACIÓN

Un lenguaje de programación es un sistema de comunicación entre el programador y la máquina (computador), creado con el único fin de programar computadoras, esto se hace partiendo del uso de instrucciones preestablecidas.

Estas instrucciones las podemos asociar al idioma usado por los humanos para comunicarse.

Un computador funciona siguiendo un conjunto de instrucciones llamadas programas, estos son los pasos que uno a uno sigue un computador para completar una tarea. (Guevara, 2008).

2.3.5. LENGUAJES DE PROGRAMACIÓN PARA LA WEB

Jabba, *et al.* (2004) expresa que: Largo ha sido el camino evolutivo del manejo de la información en internet para llegar a la utilización de sistemas e Bases de datos. La humanidad ha sido testigo de la forma en que los websites pioneros han pasado a ser maravillosos sitios interactivos de hoy en día, y en este punto existe la necesidad de saber elegir cual herramienta utilizar, ya que hay varias como ServerPages™(ASP), JavaServerPages™(JSP), Personal Home Pages(PHP), Cold Fusion, etc., y muchas veces esta elección se torna difícil, debido a que cada una tiene sus ventajas y desventajas, así como diversos usos. Por ejemplo, ASP provee el ambiente de desarrollo más fácil, mientras que JSP requiere el conocimiento de Java y Servlets. PHP es una tecnología popular basada en una fuente de desarrollo abierta. ASP se acomoda mejor para sitios pequeños que tengan entre 100-500 consultas por día, mientras que JSP es capaz de manejar entre 100 y un millón de consultas por día.

2.3.5.1. Hypertext Pre-Procesador (PHP)

En concordancia con Chaves, *et al.* (2011) PHP es un lenguaje creado en el año 1995 por PHP Group, PHP significa Hypertext Pre-processor aunque inicialmente se llamó “Personal Home Page”. Es un lenguaje de script que se interpreta en el lado del servidor, se usa para la generación de páginas web dinámicas,

embebidas en páginas HTML y ejecutadas en el servidor. Para su funcionamiento necesita tener instalado un servidor de Apache. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas. Los archivos cuentan con la extensión (php).

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. A partir de PHP 5 incorpora un modelo de objetos más completo. Entre sus características están la inclusión de la visibilidad, las clases abstractas y clases y métodos finales, manejo de excepciones, interfaces, clonación y tipos sugeridos. Forma parte de las tecnologías LAMP (Linux Apache, MySQL, PHP/Python/Perl), siendo esta plataforma una de las más extendidas en cuanto software open source para desarrollo web se refiere (Acosta, *et al.* 2012).

2.3.5.2. Active Server Pages (ASP)

Es un Lenguaje creado por Microsoft ASP lo que es en inglés “Active Server Pages”, al igual que el PHP sirve para crear páginas web dinámicas. Para que éste funcione de forma correcta necesita tener instalado los servicios de Internet Information Server (IIS). Su código se ejecuta del lado del servidor. Los archivos cuentan con la extensión (asp) (Chaves, *et al.* 2011).

Citando a Rosero *et al.* (2009), que expresa que ASP.NET es la versión .NET de la clásica tecnología ASP, esta tecnología permite implementar páginas web dinámicas. Una de las características sobresalientes de ASP.NET y que es aplicable para el diseño de aplicaciones que usan base de datos, es el trabajo con ambientes desconectados, el énfasis de esto es no mantener una conexión perenne a la base de datos, pues esto consume recursos del servidor, sino hacer una sola conexión para traer los datos inicialmente y luego desconectarse, y que la manipulación sea, a partir de allí, de manera desconectada.

2.3.5.3. Java Server Pages (JSP)

Es una tecnología web basada en la plataforma Java, ambas creadas por Sun Microsystems Inc. Lo interesante de esta tecnología es su propuesta de crear aplicaciones que sean independientes del servidor y de la plataforma, así como lo logro Java, y que a la vez permite rapidez en el desarrollo de las aplicaciones, se ejecuta del lado del servidor. JSP lo que es en inglés “Java Server Pages”. (Rosero, 2009).

Chaves, *et al.* (2011) considera que comparte características similares a las de ASP.NET, ya que fue desarrollado para la creación de aplicaciones Web potentes. JSP tiene un motor de páginas basado en los servlets de Java. Para su funcionamiento JSP necesita tener instalado un servidor Tomcat.

Analizando lo expuesto y conociendo las herramientas presentadas plantea el desarrollo del sistema bajo tecnologías Windows en el lenguaje de desarrollo ASP, ya que es una herramienta que ofrece estabilidad, escalabilidad y permite realizar gran cantidad de consultas opciones.

2.3.6. ENTORNOS DE DESARROLLO INTEGRADO (IDE)

Un Ambiente Integrado de Desarrollo es un software que provee todas las facilidades para que un programador, a través de él, pueda desarrollar otros programas. Pero no todos los ambientes están adecuados para el desarrollo de un tipo de software en particular, así surge la necesidad de evaluar la calidad de los mismos al momento de seleccionar uno (Naranjo, *et al.* 2013).

En la actualidad existe una gran cantidad del software que permiten diseñar rápidamente aplicaciones, algunos de los cuales se detallan a continuación:

2.3.6.1. NETBEANS

Es un entorno de desarrollo creado por Sun Microsystems. Es una herramienta en la que se puede escribir, compilar, depurar y ejecutar programas escritos en Java, aunque también puede servir para cualquier otro lenguaje de programación. Es de resaltar que NetBeans IDE es un producto de uso libre, gratuito y no tiene restricciones de uso (Chaves, *et al.* 2011):

2.3.6.2. CODECHARGE STUDIO

Es una solución que permite crear visualmente aplicaciones Web con bases de datos, generando una cantidad mínima de código. Posee soporte para casi todas las bases de datos, servidores web y tecnologías web.

Su entorno gráfico de usuario y su generador de código permiten construir rápidamente aplicaciones web mediante la generación de código robusto, escalable de programación. CodeCharge Studio permite generar código en los lenguajes ASP.NET (C #), ASP, PHP, Java Servlets, JSP, ColdFusion y Perl.

El Application Builder incluido en éste programa permite convertir rápidamente una base de datos (MS Access, MS SQL, MySQL, Oracle, etc) en una aplicación web de trabajo con manejo de usuarios de inicio de sesión protegida, permite también la búsqueda de información de forma fácil en la base de datos y funciones de actualización de registros (Chaves, *et al.* 2011).

2.3.6.3. SUBLIME TEXT

Es un editor de texto con todas las características ideales para editar archivos de texto locales. Tiene muchas características incorporadas para ayudar en la edición

de código, como resaltado de sintaxis, auto sangría, reconocimiento de tipo de archivo, una barra lateral de archivos / carpetas a mano para fácil edición de múltiples archivos en un directorio, guía de macros para automatizar tareas repetitivas, y las pestañas para ver y editar varios archivos al mismo tiempo. Con gran cantidad de características, utilizar este editor puede aumentar su productividad.

Entornos (IDE), como Visual Studio y Eclipse. Estos IDEs ciertamente tienen su lugar, pero la simplicidad de Sublime Text 2 puede ser preferible en otros escenarios. Además de las muchas características integradas útiles, Sublime Text fue construido desde cero para ser extensible y la comunidad Sublime Text tiene tomado nota de ello. Ya hay muchas extensiones útiles que incluyen interfaces para sistemas de control de versiones como git, paquetes fragmento de jQuery y PHP, y resaltado de sintaxis para lenguajes, hojas de estilo CSS (Haughee, 2013).

2.3.6.4. VISUAL STUDIO.NET

Microsoft Visual Studio es un IDE (Entorno de desarrollo integrado para sistemas operativos Windows. Visual Studio punto Net Soporta varios lenguajes de programación como lo son Visual Visual C#, Visual J#, C++, ASP.NET y Visual Basic .NET. Actualmente se han desarrollado las extensiones necesarias para muchos otros lenguajes de programación.

Visual Studio permite a los desarrolladores de software crear aplicaciones de escritorio, aplicaciones de tipo cliente servidor, aplicaciones y servicios Web, y también aplicaciones para dispositivos móviles que funcionen con sistema operativo Windows (Chaves, *et al.* 2011):

2.3.7. FRAMEWORK

Vanegas (2013) expresa que uno de los grandes retos de la programación orientada a objetos es facilitar la combinación y el acceso a cualquier tipo de información, como una característica integrada a un lenguaje de programación.

Escobar *et al.* (2012) define un framework como un conjunto de mejores prácticas, métodos y herramientas. En el desarrollo de software, un framework o infraestructura digital, es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, que puede servir de base para la organización y desarrollo de software. Típicamente, puede incluir soporte de programas, bibliotecas, y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Se han utilizado dos framework para implementar la funcionalidad del sistema y para el diseño de la parte visual. A continuación se describirán y se especificarán las funciones principales que se han utilizado.

2.3.7.1. ADO .NET ENTITY FRAMEWORK

Cibertec (s.f.) citado por Mindy y Erixc (2011) definen ADO.net como un conjunto de componentes del software que pueden ser usados por los programadores para acceder a datos y a servicios de datos. Es frecuentemente usado por los programadores para modificar y acceder a datos almacenados en un sistema Gestor de bases de datos relacionales, así como también a datos en fuentes no relacionales.

ADO.NET Entity Framework es una mejora a ADO.NET creada por Microsoft, que dota a los desarrolladores de un mecanismo adicional para acceder a los datos y trabajar con los resultados, además de los Data Readers y Data Sets utilizados en versiones anteriores.

Utilizando esta tecnología se pueden desarrollar aplicaciones de escritorio, aplicaciones cliente-servidor, aplicaciones web, utilizando herramientas complementarias que nos ayuden a lograr estos propósitos como ASP NET, Silverlight o Windows Communication Foundation VCF).

Entity Framework (EF) abstrae la estructura de la base de datos y todo el acceso y almacenamiento de datos se realiza frente a un modelo conceptual de datos que refleja los objetos de negocio. Los ítems usados para representar el modelo conceptual se conocen como Entidades (Suárez, 2011).

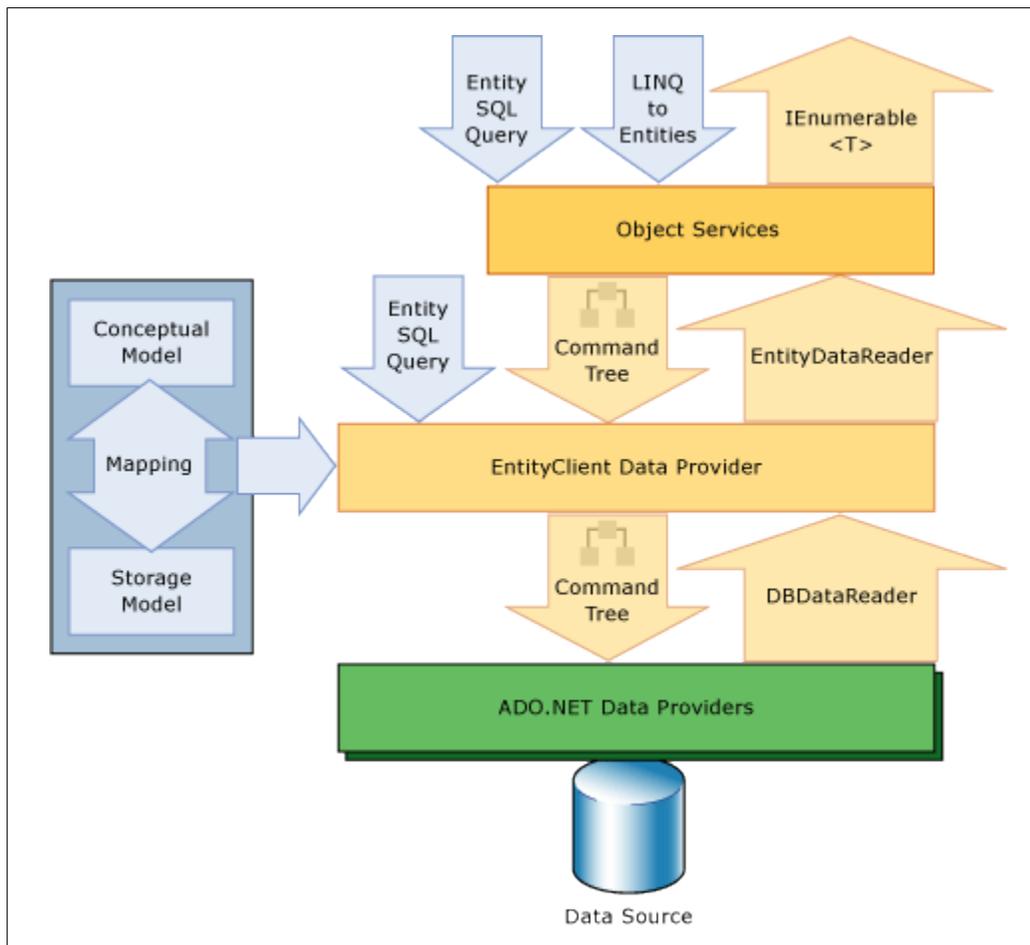


Figura 2.1. Arquitectura ado.net entity framework

2.3.7.2. BOOTSTRAP FRAMEWORK

Olmos (2013) considera que Twitter Bootstrap es una colección de herramientas de software libre para la creación de sitios y aplicaciones web destinado a la capa visual y está basado en JavaScript y CSS. Contiene plantillas con tipografías, formularios, botones, gráficos, barras de navegación y demás componentes de interfaz, así como extensiones opcionales de JavaScript.

El objetivo de este framework es facilitar el diseño y la ordenación de la capa visual, además de disponer de multitud de tipografías, formularios, botones, gráficos, barras de navegación que nos permite generar una página web rica en componentes visuales y con una experiencia del usuario más satisfactoria. Otra característica muy importante que nos ofrece este framework es la compatibilidad con los diferentes navegadores existentes. Ya sean en ordenadores, tablets o móviles, la aplicación web se adaptará automáticamente a las necesidades adaptando la capa visual de la aplicación.

La ordenación de la capa visual en este framework se llama Scaffolding y es uno de sus atractivos más importantes, ya que divide el contenido de todos los componentes visuales, incluida la página web en sí misma, en 12 columnas verticales; pudiendo definir claramente cuantas columnas debe de ocupar cada componente y evitar así tener que configurar el posicionamiento de cada uno de ellos.

En el sistema de test on-line, el framework Bootstrap se ha utilizado para la implementación de la capa visual del panel de control y del módulo de rendimiento en línea.

2.3.8. BASES DE DATOS

Las bases de datos son masivamente utilizadas en las aplicaciones de hoy en día. Detrás de la mayoría de los sistemas informáticos, existe una base de datos así como un sistema de manejo de la misma, que permite el acceso a los datos, brindando seguridad al usuario, recuperación ante fallos, posibilidad de acceso concurrente, y mecanismos de control de concurrencia que garantizan la atomicidad de las transacciones, y la correcta modificación de los datos. Dentro de los diferentes modelos, sin lugar a dudas, el modelo relacional es el modelo que más éxito ha tenido y que más ha perdurado en el tiempo. Probablemente la razón de esto sea debido a la solidez formal del modelo en sí, como así también a la eficiencia de los sistemas de manejo de bases de datos que lo utilizan. Sin embargo, el modelo relacional puro tiene una fuerte limitación derivada de las restricciones impuestas por las formas normales: los campos de las relaciones (tablas) deben ser atómicos. Actualmente, existen dos formas de extender dichos sistemas: mediante bases de datos objeto-relacionales, o mediante la interacción con lenguajes de programación orientados a objetos que permitan la manipulación objetos persistentes (Falappa, *et al.* 2012).

2.3.9. SISTEMAS GESTORES DE BASES DE DATOS

Con el desarrollo de las distintas tecnologías asociadas a la informática como ciencia, se concede gran importancia a la gestión de la información a través de los medios electrónicos, pues perfecciona la fluidez de los procesos, la toma de decisiones y la calidad del trabajo en diferentes organizaciones. Uno de los aportes que ha traído el desarrollo de los campos de informática y electrónica ha sido el concepto de Base de Datos (BD) con formatos electrónicos, por cuanto aporta a la planificación y organización en una empresa. La idea es agrupar todos los datos relevantes en un único lugar; la expresión: "datos relevantes" hace referencia a aquella información que por su significado será necesario almacenar y conservar, ya que sustentará las actividades del negocio y la toma de decisiones

correspondientes en la empresa y el parámetro que guiará la elección de dichos datos estará dado por los objetivos que se persigan.

Un sistema gestor de bases de datos (SGBD o DBMS), es un conjunto de programas que se encargan de manejar la creación y todos los accesos a las bases de datos. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta (Rondón, *et al.* 2011).

Los sistemas gestores de bases de datos, hoy en día, constituyen la tecnología más usada en la creación de sistemas de información, por lo cual es imprescindible el desarrollo de técnicas que permitan el almacenamiento y posterior consulta de información (Reyna, *et al.* 2012).

2.3.9.1. SQL SERVER

Microsoft SQL Server es un conjunto completo de tecnologías y herramientas específicas para la empresa que permiten que el personal obtenga de la información el máximo valor al menor costo de propiedad. SQL Server es una plataforma segura, confiable y escalable para las aplicaciones cruciales de una empresa, reduce el tiempo y el coste de administración de cualquier tipo de datos. Simplifica el desarrollo de aplicaciones controladas por datos, a la vez que proporciona la información correcta a todos los usuarios, estén donde estén (Shapiro, 2009).

2.4. INGENIERÍA DE SOFTWARE

La Ingeniería de Software (IS) es la disciplina o área de la informática que ofrece métodos y técnicas para el desarrollo y mantenimiento de software de calidad. A pesar de todos los esfuerzos realizados, en las sucesivas etapas del desarrollo y en las actividades de gestión de proyectos de software, ocurren problemas como:

carencia de fiabilidad, necesidad de mantenimiento permanente, etc. Las dinámicas de los procesos de desarrollo de software exigen una actualización constante de técnicas y tecnologías. Esta tarea es favorecida por la utilización de herramientas CASE que asisten a todos los involucrados en la construcción del software (Daniele, *et al.* 2010).

El desarrollo de software es una actividad compleja que requiere la integración de factores técnicos, gerenciales y organizacionales, ha sido considerado tanto un arte como una ciencia Lawrence (2002) citado por Anaya (2012). Como ciencia, esta disciplina se fundamenta en la aplicación de prácticas de ingeniería que permiten estimar, medir y evaluar el proceso de desarrollo, de manera repetitiva y controlada.

Como arte, el desarrollo de software requiere producir resultados que reflejen ingenio y habilidad para seleccionar, diseñar y construir el producto software que mejor satisfaga los requerimientos de una organización (Anaya, 2012).

A partir de los modelos verbales es posible construir esquemas conceptuales, que son diagramas que permiten representar gráficamente los datos y funciones asociados con el problema para realizar el desarrollo del software (Jaramillo, 2012).

2.4.1. INGENIERÍA DE REQUISITOS

Hechavarría y Herrera (2012) definen que la Ingeniería de Requisitos describe técnicas que permiten la captura requisitos de software, la recopilación de la información y la identificación de la adecuada para usarlas.

El desarrollo de software inicia con una serie de entrevistas realizadas a los usuarios potenciales con el fin de determinar los requisitos del software; como resultado de las entrevistas se obtienen modelos verbales en lenguaje natural. A

partir de los modelos verbales es posible construir esquemas conceptuales, que son diagramas que permiten representar gráficamente los datos y funciones asociados con el problema para realizar el desarrollo del software (Zapata y Arango, 2012).

La ingeniería de requerimientos (IR) incluye el conjunto de tareas que conducen a comprender cuál será el impacto del software sobre el negocio, que es lo que el cliente quiere y cómo interactúan los usuarios finales con el software. La IR, como todas las demás actividades de la ingeniería del software, debe adaptarse a las necesidades del proceso, el proyecto y las personas que realizan el trabajo. Desde la perspectiva del proceso del software, la ingeniería de requisitos es una acción de la ingeniería de software que comienza durante la actividad de comunicación y continúa en la actividad de modelado.

El proceso de la ingeniería de requisitos se lleva a cabo a través de siete distintas funciones: inicio, obtención, elaboración, negociación, especificación, validación y gestión (Pressman, 2010).

2.4.1.1. UML

En el desarrollo de sistemas de información existe el interés por mejorar la productividad y crear productos de software de alta calidad. Para ello, es indispensable establecer un entorno disciplinado y estructurado que genere una ventaja competitiva frente a las demás organizaciones y en el cual predomine el trabajo en equipo, para evitar resultados impredecibles.

El UML es el lenguaje de mayor difusión que la OMG2 ha convertido desde 1997 en el estándar para definir, organizar y visualizar los elementos que configuran la arquitectura de una aplicación, sea o no de software, es una especificación de notación orientada a objetos, el cual se compone de diferentes diagramas, los

cuales representan las diferentes etapas del desarrollo del proyecto (Quintero, *et al.* 2012).

2.4.1.2. DIAGRAMA DE CASOS DE USO

El diagrama de casos de uso es importante en el desarrollo de aplicaciones de software para capturar los requisitos funcionales y para manejar la complejidad de sistemas robustos. En la especificación de la Superestructura del Unified Modeling Language UML, el diagrama de casos de uso se define como el “diagrama que muestra las relaciones entre los actores y el sujeto (sistema) y los casos de uso, describe los requisitos funcionales del sistema en términos de las secuencias de acciones. Se presentan los siguientes elementos de su especificación:

- Casos de uso: son las especificaciones de un conjunto de acciones realizadas por el actor sobre el sistema.
- Actores: son los roles que los usuarios desempeñan respecto del sistema y que emplean los casos de uso.
- Relaciones: identifican la comunicación existente entre actores y casos de uso.

Las principales ventajas de utilizar este diagrama, son:

- La captura de los requisitos funcionales desde el punto de vista del usuario.
- La utilización de los casos de uso para educir y documentar los requisitos funcionales.
- El manejo de la complejidad en sistemas robustos, descomponiendo el problema en funciones más simples (Zapata y Tamayo, 2009).

La representación gráfica de los elementos del diagrama de casos de uso se puede apreciar en la figura 2.2.

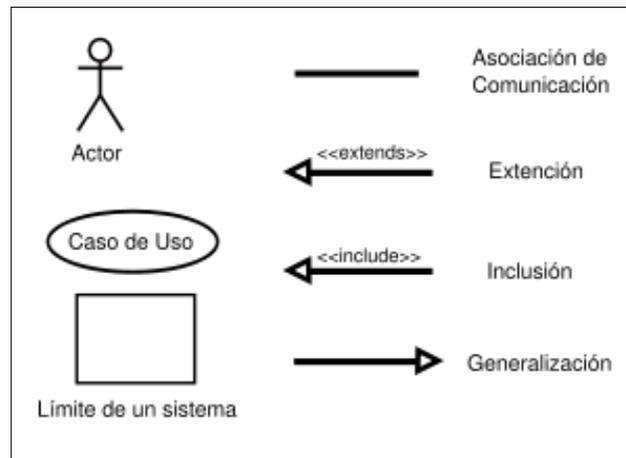


Figura 2.2. Elementos del diagrama de caso de uso.

2.4.2. MODELOS DE PROGRAMACIÓN

La realización de Sistemas de Información se ha venido desarrollando en base a técnicas de programación, principalmente: la programación estructurada, luego en combinación utilizando la programación por eventos. Actualmente se pudiera decir que se ha llegado a una madurez con la potencialidad de la programación orientada a objetos por la ventaja en la reutilización de código. En adición a ellas, se cuenta actualmente con la programación en n capas que hace uso de la programación orientada a objetos: la cual consiste en separar el código fuente según el rol, responsabilidad y funcionalidad; por ende el desarrollo es más rápido. Y resulta más fácil el darle mantenimiento al Sistema (Moquillaza, *et al.* 2010).

2.4.2.1. PROGRAMACIÓN EN N CAPAS

Álvarez y Montes, *et al.* (2010) manifiestan que la programación en capas, no es nada más que un estilo de programación donde su objetivo primordial es separar diferentes aspectos de desarrollo, separando la lógica de negocio de la lógica de diseño; para luego unirlos ya sea en código o en tiempo de ejecución, de aquí salen la programación en dos capas, tres capas, cuatro capas, n-capas.

2.4.2.1.1. TIPOS DE CAPAS

Capa de presentación: Es la responsable de la presentación visual de la aplicación. La capa de presentación enviará mensajes a los objetos de esta capa de negocios o intermedia. La cual o bien responderá entonces directamente o mantendrá un diálogo con la capa de la base de datos, la cual proporcionará los datos que se mandarían como respuesta a la capa de presentación (Moquillaza, *et al.* 2010).

Se puede concluir que es la capa del proyecto responsable de la interacción directa con el usuario, en esta etapa se deben tomar las consideraciones necesarias para garantizar que la distribución de la información permita el fácil uso.

Capa de negocio: Es la responsable del procesamiento que tiene lugar en la aplicación. Contendrá objetos definidos por clases reutilizables. Estos objetos se suelen llamar objetos de negocios y son los que contienen la gama normal de constructores, métodos para establecer y obtener variables, métodos para establecer y obtener variables, métodos que llevan a cabo cálculos y métodos normalmente privados, en comunicación con la capa de datos (Moquillaza, *et al.* 2010).

Capa de datos: Esta capa se encarga de acceder a los datos, es la encargada de preocuparse por la conexión y gestión de los datos, Es aquí donde se implementan las conexiones al servidor y a la base de datos del sistema. Esta capa envía la información directamente a la capa de reglas de negocio para que

sea procesada e ingresada en objetos según se necesite, esta acción se denomina encapsulamiento (Alvarez y Montes, 2010).

2.4.3. PROGRAMACIÓN ORIENTADA A OBJETOS

Según Grady (s.f.) citado por Moquillaza, *et al.* (2010), la programación orientada a objetos es un método de implementación en el que los programas se organizan como colecciones cooperativas de objetos, cada uno de los cuales representa una instancia de alguna clase y cuyas clases son todas miembros de una jerarquía de clases unidas mediante relaciones de herencia”.

Los programas orientados a objetos constan de objetos que se comunican entre sí a través de mensajes.

2.4.4. METODOLOGÍAS DE DESARROLLO AGILES

Un método de desarrollo de software se dice que es un método ágil de desarrollo de software cuando se centra en las personas, es orientado a la comunicación, flexible (listo para su adaptación a la espera de un cambio inesperado en cualquier momento), rápida (estimula el rápido e iterativo desarrollo del producto en versiones pequeñas), eficiente (se centra en acortar los plazos y costes y en la mejora de la calidad), adaptable (reacciona adecuadamente a los cambios esperados e inesperados), y aprende (centrado en la mejora durante y después del desarrollo del producto) (Avila y Meneses, 2012).

La constante innovación tecnológica hace que cada vez sea más necesaria la aplicación de nuevas metodologías adaptadas a los nuevos tiempos, esto sin embargo, descuida el estudio de las metodologías más antiguas y que sin lugar a dudas constituyen la base de la Ingeniería de Software porque exhiben buenas prácticas que pudieran ser utilizadas. Todas las metodologías son, en esencia,

bien intencionadas. Obviamente, las más modernas responden a problemas, necesidades y tendencias actuales (Heredia, *et al.* 2011).

2.4.4.1. PROGRAMACIÓN EXTREMA

La Programación Extrema (XP), es una metodología ligera de desarrollo de software que se basa en la simplicidad, la comunicación y la realimentación o reutilización del código desarrollado. La metodología consiste en una programación rápida o extrema, utilizadas para proyectos de corto plazo (Heredia, *et al.* 2011).

A continuación detallamos algunas características del desarrollo de la metodología XP mencionados por Highsmith citado por Heredia (2011).

- Los diseñadores y programadores se comunican efectivamente con el cliente y entre ellos mismos.
- Los diseños del software se mantienen sencillos y libres de complejidad o pretensiones excesivas.
- Se obtiene retroalimentación de usuarios y clientes desde el primer día gracias a las baterías de pruebas.
- El software es liberado en entregas frecuentes tan pronto como sea posible.
- Los cambios se implementan rápidamente tal y como fueron sugeridos.
- Las metas en características, tiempos y costos son reajustadas, permanentemente en función del avance real obtenido.

Alvarez (2005) citado por Mindy y Erick (2011) explica la metodología XP como una metodología de desarrollo de software orientada a agilizar el tiempo de desarrollo por medio del trabajo en pareja de los desarrolladores y un alto involucramiento del cliente como proveedor de requerimientos durante todo el proceso.

Una de las Metodologías Ágiles más importante y reconocida es la Programación Extrema (XP), para aquellos proyectos de software donde el cambio en los requerimientos es la norma. Por tal motivo reúne un conjunto de prácticas sencillas ya conocidas, pero que en este caso son llevadas a cabo conjuntamente y en forma extrema (Kasiak y Godoy, 2012).

2.4.4.1.1. FASES DE LA METODOLOGÍA DE DESARROLLO EXTREME PROGRAMMING XP

La metodología extreme programming se compone de varias fases; planificación del proyecto, diseño, codificación, pruebas. Se eligió esta metodología por ser adaptable a las necesidades del sistema y sin seguir un régimen estricto.

2.4.5. INGENIERÍA WEB

Hoy día, con el devenir de la alta tecnología del siglo XXI, las compañías se encuentran desarrollando productos y servicios cada vez más complejos, capaces de satisfacer las necesidades de los clientes. Es por ello que se impone un nuevo reto, los productos o servicios deben contar con la más alta calidad, pues a mayor calidad, mayor será la demanda del mismo (Jiménez, *et al.* 2012).

La Ingeniería Web (Web) aplica sólidos principios científicos, de ingeniería y de administración, y enfoques disciplinados y sistemáticos para el desarrollo, despliegue y mantenimiento exitoso de sistemas y aplicaciones basados en Web de alta calidad (Nieves, 2009).

La Ingeniería Web tiene como objetivo la aplicación y desarrollo de enfoques de la Ingeniería de Software a las aplicaciones Web. Existen varias características que

ubican a las aplicaciones Web en una posición especial dentro de la Ingeniería de Software. Algunas de ellas según Oliveros *et al.* (2011) son:

- La importancia del contenido y de la presentación en un sitio Web hacen necesaria una estrecha colaboración de un equipo en el que coexisten programadores, especialistas en contenido y diseñadores gráficos.
- Existen, además de las tradicionales aplicaciones centradas en documentos muchas otras. Por citar sólo algunas, podemos mencionar las aplicaciones interactivas, transaccionales, sociales y colaborativas.
- Los requerimientos de calidad se vuelven más complejos, ya que hay que considerar la calidad del contenido (consistencia, confiabilidad, actualización, relevancia entre otros.) Asimismo es necesario tener en cuenta la calidad de la estructura de hipertexto que generalmente subyace en un sitio Web.
- El contexto técnico es también complejo. La multiplicidad de plataformas desde las cuales se puede acceder una aplicación Web cambia tanto que es difícil para un usuario normal tener un panorama general. Esto hace que el tradicional “mantra” que se suele invocar cuando se habla de requerimientos (hay que concentrarse en “qué” y no en “cómo”) se vuelve impracticable en el caso de aplicaciones Web por la especial interacción que hay entre los requerimientos y la arquitectura.
- Finalmente, la localización geográfica del usuario requiere de las aplicaciones capacidad para adaptarse a diversos contextos locales por ejemplo, diferencias de hora, idioma, cultura, etc.

2.4.5.1. PROCESOS DE LA IWEB (INGENIERÍA WEB)

La inmediatez, evolución y crecimiento continuos, son características de las aplicaciones Web, esto nos lleva a un proceso incremental y evolutivo, que permite que el usuario se involucre activamente, facilitando el desarrollo de productos que se ajustan a sus requerimientos.

Pressman (s.f.) citado por Nieves (2009) enumera siete actividades que forman parte del proceso de la Web y que son aplicables a cualquier WebApp independientemente de su tamaño y complejidad. Las mismas son:

1. La Formulación identifica objetivos y establece el alcance de la primera entrega.
2. La Planificación genera la estimación del coste general del proyecto, la evaluación de riesgos y el calendario del desarrollo y fechas de entrega.
3. El Análisis especifica los requerimientos e identifica el contenido.
4. La Modelización se compone de dos secuencias paralelas de tareas. Una consiste en el diseño y producción del contenido que forma parte de la aplicación. La otra, en el diseño de la arquitectura, navegación e interfaz de usuario. Es importante destacar la importancia del diseño de la interfaz. Independientemente del valor del contenido y servicios prestados, una buena interfaz mejora la percepción que el usuario tiene de éstos.
5. En la Generación de Páginas se integra contenido, arquitectura, navegación e interfaz para crear estática o dinámicamente el aspecto más visible de la aplicación: las páginas.

6. El Test busca errores a todos los niveles: contenido, funcional, navegacional, rendimiento, etc. El hecho de que las aplicaciones residan en la red, y que interoperen en plataformas muy distintas, hace que el proceso de test sea especialmente difícil.
7. Finalmente, el resultado es sometido a la Evaluación del Cliente

Una de las tareas colaterales que forman parte del proceso es el Control y Garantía de la Calidad. Este proceso, que se cumple en la ingeniería de software tradicional, implica actividades tales como: establecimiento y supervisión de estándares, revisiones técnicas formales, análisis, seguimiento y registro de informes, etc, son igualmente aplicables a la Ingeniería Web. Sin embargo, en la Web toman especial relevancia para valorar la calidad, aspectos como: usabilidad, funcionalidad, fiabilidad, seguridad, eficiencia y mantenibilidad.

2.4.5.2. MÉTODOS DE LA IWEB

Nieves (2009) manifiesta que los métodos de la Ingeniería Web definen las etapas y actividades necesarias para efectuar la construcción completa de una aplicación Web. El principio subyacente en todos ellos es que una aplicación Web debe desarrollarse partiendo de una descripción precisa en forma de un esquema conceptual que se transforma a una representación software, mediante un conjunto de correspondencias entre las abstracciones conceptuales que constituyen su esquema conceptual y los componentes software.

En menor o mayor medida y a veces con diferentes nombres o sub-fases, la mayoría de los métodos coinciden en las siguientes etapas:

1. Diseño Conceptual: Trata de la especificación del dominio del problema, a través de la definición de datos y sus relaciones.

2. Diseño Navegacional: Establece los caminos de acceso a la información y sus permisos de visibilidad.
3. Diseño de la presentación o diseño de Interfaz: Define cómo se muestra la información en la interfaz de usuario.
4. Implementación: Es la construcción del software a partir de los artefactos generados en las etapas previas.

CAPITULO III. DESARROLLO METODOLÓGICO

El presente capítulo describe las técnicas y metodologías utilizadas en la tesis con la finalidad de demostrar que los resultados están respaldados técnicamente, el sistema de generación de test está pensado para que la coordinación de acreditación de la ESPAM MFL diseñe evaluaciones que posteriormente serán realizadas por estudiantes con el afán de ponderar y evidenciar los déficits académicos existentes y tomar los correctivos necesarios con miras a la acreditación.

3.1. METODOLOGÍA DE INVESTIGACIÓN

Para la ejecución técnica de la presente tesis fue necesario emplear los métodos inductivo y deductivo, como metodologías de investigación, y el proceso XP, como metodología informática.

3.1.1. MÉTODO INDUCTIVO

Se empleó en la observación de los hechos para su registro, clasificación y estudio, con la finalidad de entender el proceso y obtener razonamientos sobre el manejo actual de las evaluaciones.

Para tener las bases conceptuales referentes a procesos de evaluación mediante plataformas virtuales, el autor investigó conceptos generales entorno al e-learning y a los generadores de test, conceptos que fueron necesarios conocer, ya que están implícitamente relacionados al tema de tesis y permitieron inferir sobre temáticas relacionadas.

3.1.2. MÉTODO DEDUCTIVO

Se aplicó en el análisis de la información, obtenida luego de la entrevista en la toma de requerimiento lo que permitió inferir datos necesarios y consideraciones durante el desarrollo del software.

3.2. METODOLOGÍA INFORMÁTICA

La metodología que el autor implementó fue el Proceso XP (Programación extrema) considerada en los métodos de desarrollo ágil, el mismo que consta de cuatro fases: planificación, diseño, codificación y pruebas.

3.2.1. FASE DE PLANEACIÓN

Consistió en el inicio del desarrollo del proyecto y se fundamentó en la necesidad e importancia de las tareas de toma de requerimientos que fueron determinantes para el éxito o fracaso del proyecto. Esta fase contempló la entrega de información por parte de la comisión de evaluación y permitió recopilar los datos necesarios para establecer los requisitos y las funcionalidades del sistema, se conocieron estos datos mediante entrevista con el ingeniero Ángel Vélez coordinador del comité de evaluación institucional de la Escuela Superior Politécnica Agropecuaria de Manabí de la cual se obtuvo el documento de entrega de requerimientos como lo muestra el anexo 1.

3.2.1.1. ESTUDIO DE FACTIBILIDAD

El Sistema de Educación Superior conformado por las Escuelas Politécnicas y Universidades del país desarrollan todas sus actividades orientadas a acreditarse como universidades de excelencia académica. La Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López no es la excepción, la coordinación

de acreditación institucional, es el departamento encargado de realizar actividades enfocadas a la acreditación externa.

Para el estudio de factibilidad se levantó informaciones referentes a las plataformas (software) existentes en la universidad, y se listaron las que el sistema podría utilizar, lo que evidenció la existencia de herramientas detalladas a continuación:

- Servidor con Sistema Operativo Windows
- Servidores:
 - IIS de Windows (ASP-C#)
 - Apache de Linux (PHP)
- Gestores de base de datos:
 - SQL Server (Windows, Software Propietario)
 - Mysql (Linux, Software Libre)

Con esta información, el sistema desarrollado se basó en tecnologías y herramientas existentes en la universidad con la finalidad de reducir costos y tiempos de desarrollo y adquisición de equipos, facilitando la integración con la base de datos de matriculación existente, por lo cual se hizo necesario trabajar en coordinación con la unidad de producción de software encargada de la administración de la base de datos en mención.

3.2.1.2. OBTENCIÓN Y ANÁLISIS DE REQUERIMIENTOS

En entrevista con el coordinador de acreditación se determinaron los siguientes requerimientos:

- Ingresar, editar, eliminar y consultar sub áreas relacionadas a una asignatura específica.
- Ingresar, editar, eliminar y consultar Temas relacionadas a una sub área específica.
- Ingresar, editar, eliminar y consultar Preguntas relacionadas a un tema específico.

- Las preguntas pueden ser imágenes o texto dependiendo del tipo.
- Crear, editar, eliminar, consultar pruebas y generar reportes de las mismas.
- Relacionar aleatoriamente según los temas los reactivos que se utilizaran en la prueba.
- Permitir que un estudiante realice la prueba solo una vez.
- Validar los accesos no autorizados al sistema
- Permitir obtener reportes relacionados a una evaluación con datos estadísticos.

3.2.1.3. VALIDACIÓN DE REQUERIMIENTOS

Fue fundamental representar los actores en la entrevista informal y las tareas que realizarían los mismos según sus privilegios mediante los diagramas de casos de uso.

ACTOR	FUNCIÓN
 <p>ADMINISTRADOR</p>	<ul style="list-style-type: none"> • Iniciar sesión. • Ingresar, editar, eliminar y consultar áreas. • Ingresar, editar, eliminar y consultar sub áreas. • Ingresar, editar, eliminar y consultar temas • Ingresar, editar, eliminar y consultar Preguntas • Diseñar, editar, eliminar y consultar pruebas. • Generar reportes. • Cerrar sesión.
 <p>ESTUDIANTE</p>	<ul style="list-style-type: none"> • Identificación • Dar prueba. • Visualizar resultado. • Cerrar sesión.

Figura 3.3 Identificación de actores del sistema.

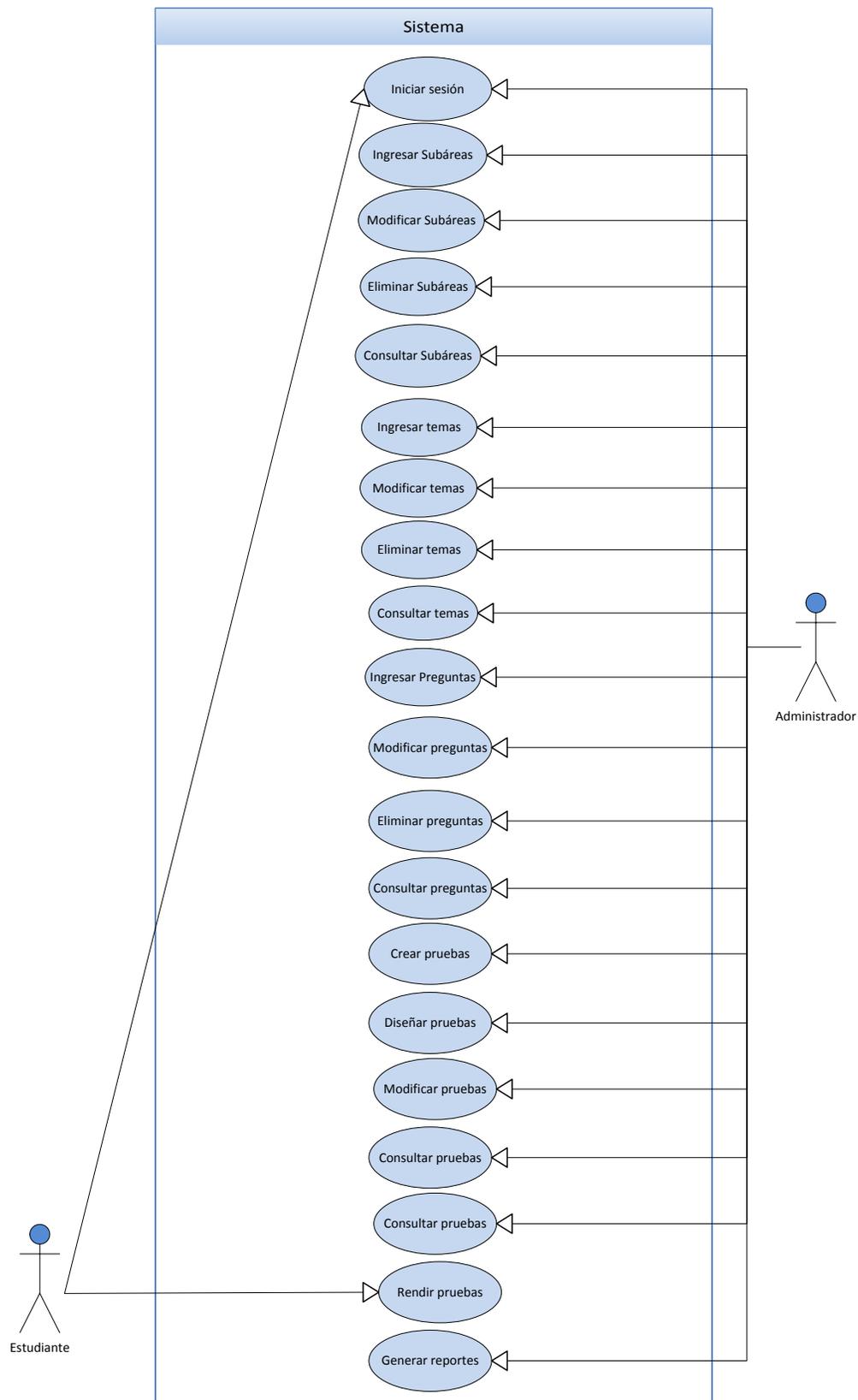


Figura 3.4. Diagrama general de funciones del sistema

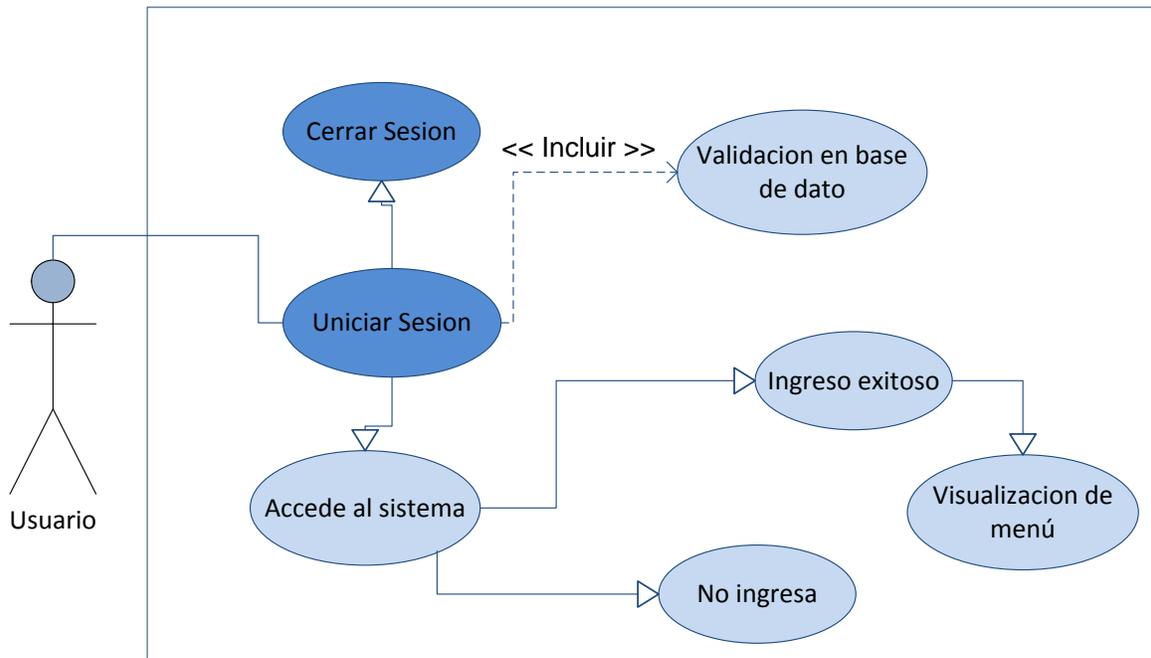


Figura 3.5. Diagrama de validación de usuario.

Cuadro 3.1 Casos de uso validación de usuario, Iniciar sesión.

Caso de uso	Iniciar Sesión	
Actor(es)	Administrador Estudiantes	
Descripción	Representa el proceso de inicio de sesión de usuarios previo ingreso a la interfaz de validación inicial del sistema.	
Flujo	<i>EVENTOS ACTOR</i>	<i>EVENTOS SISTEMA</i>
	1. Estar en la sección de ingreso de usuario y contraseña del sistema.	
	2. Llenar el formulario de	2. Mostrar formulario de inicio

	credenciales de usuarios.	de sesión.
	3. Enviar datos ingresados para validación.	3. Validar credenciales de usuario en base de datos
	4. Recibir confirmación de inicio de sesión exitoso.	4. Mostrar ventana de inicio de sesión exitoso.
Precondición	El usuario a ingresar debe previamente tener asignado un usuario y clave por parte de la institución. Ser estudiante de la institución en el caso de inicios de sesión para rendir evaluaciones.	
Pos condición	El usuario accederá al sistema y tendrá privilegios específicos dependiendo del tipo de usuario.	
Presunción	Sistema y servidor de base de datos disponibles.	

Cuadro 3.2 Casos de uso validación de usuario, Cerrar sesión.

Caso de uso	Cerrar Sesión	
Actor(es)	Administrador Estudiantes	
Descripción	Representa el proceso de cierre de sesión de usuarios.	
Flujo principal	<i>EVENTOS ACTOR</i>	<i>EVENTOS SISTEMA</i>
	1. Estar autenticado en el sistema.	
	2. Estar en la sesión visible de	2. Mostrar opción de salir del

	salida del sistema	sistema.
	3. Seleccionar el botón de salir de sesión.	3. Finaliza sesión, elimina información guardada en cookies y redirección a la página principal.
Precondición	El usuario debe estar autenticado en el sistema, presionar el botón salir del sistema	
Presunción	Sistema y servidor de base de datos disponibles.	

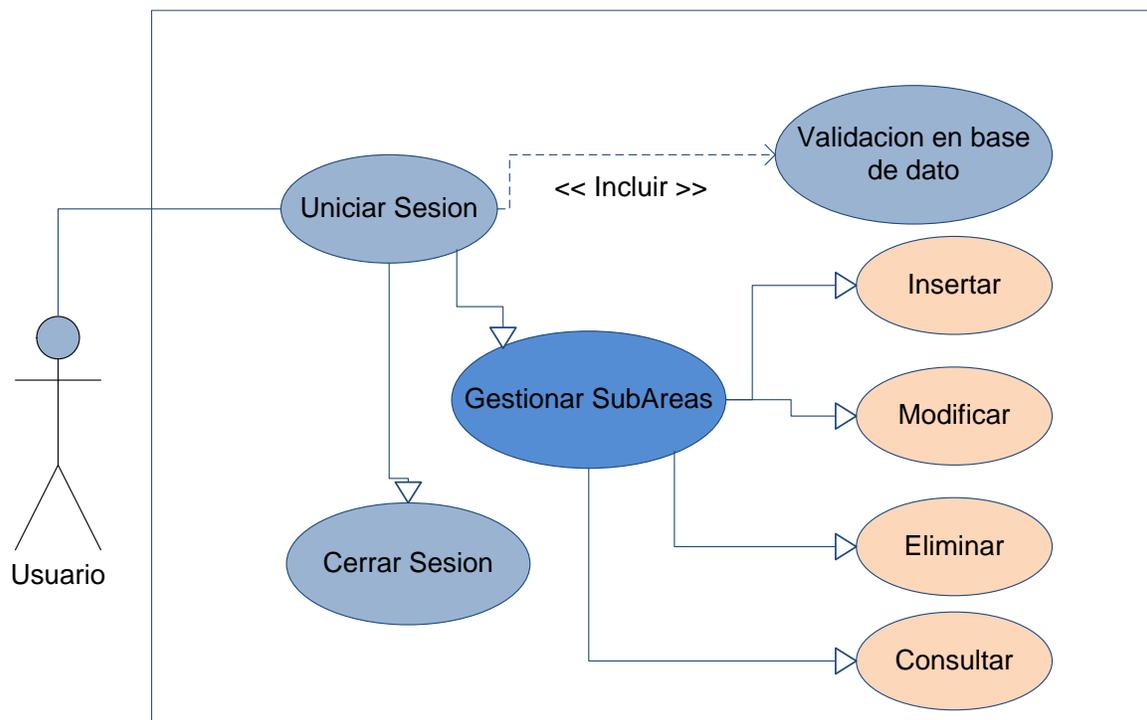


Figura 3.6. Caso de uso. Gestión de Sub Áreas.

Cuadro 3.3 Caso de uso Gestión de Sub Áreas

Caso de uso	Gestión de Sub Áreas
-------------	-----------------------------

Actor(es)	Administrador	
Descripción	Representa los procesos de insertar, modificar, eliminar y consultar sub Áreas de un test por parte del Administrador, previo validación de ingreso al sistema.	
Flujo	<i>EVENTOS ACTOR</i>	<i>EVENTOS SISTEMA</i>
	1. Estar autenticado en el sistema.	
	2. Estar en la sección del menú.	2. Mostrar menú de opciones
	3. Escoger la opción a realizar	3. Mostrar los elementos correspondientes a la acción seleccionada.
	4. Realiza las tareas correspondientes a la acción a realizar.	
	5. Recibir confirmación de acción exitosa.	5. Mostrar ventana de confirmación de acción.
Precondición	Estar autenticado en el sistema.	
Presunción	Sistema y servidor de base de datos disponibles.	

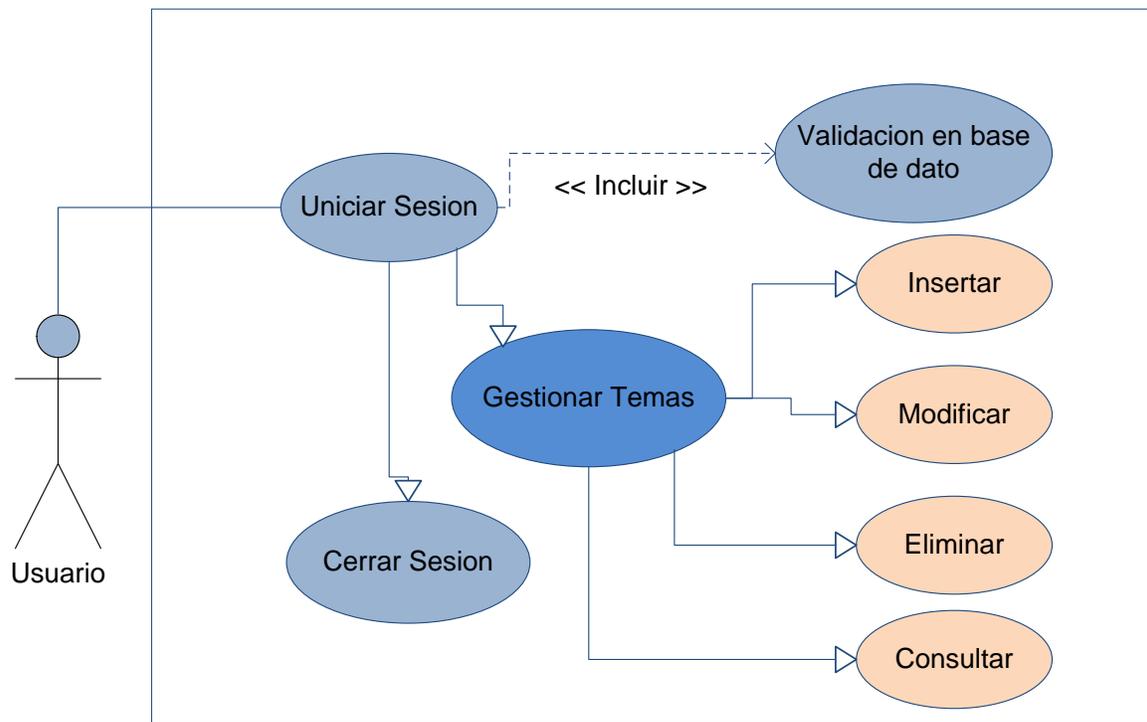


Figura 3.7. Diagrama de Gestión de Temas.

Cuadro 3.4 Caso de uso Gestión de Temas

Caso de uso	Gestión de Temas	
Actor(es)	Administrador	
Descripción	Representa los procesos de insertar, modificar, eliminar y consultar temas de un test por parte del Administrador, previo validación de ingreso al sistema.	
Flujo	<i>EVENTOS ACTOR</i>	<i>EVENTOS SISTEMA</i>
	1. Estar autenticado en el sistema.	
	2. Estar en la sección del menú.	2. Mostrar menú de opciones

	3. Escoger la opción a realizar	3. Mostrar los elementos correspondientes a la acción seleccionada.
	4. Realiza las tareas correspondientes a la acción a realizar.	
	5. Recibir confirmación de acción exitosa.	5. Mostrar ventana de confirmación de acción.
Precondición	Estar autenticado en el sistema.	
Presunción	Sistema y servidor de base de datos disponibles.	

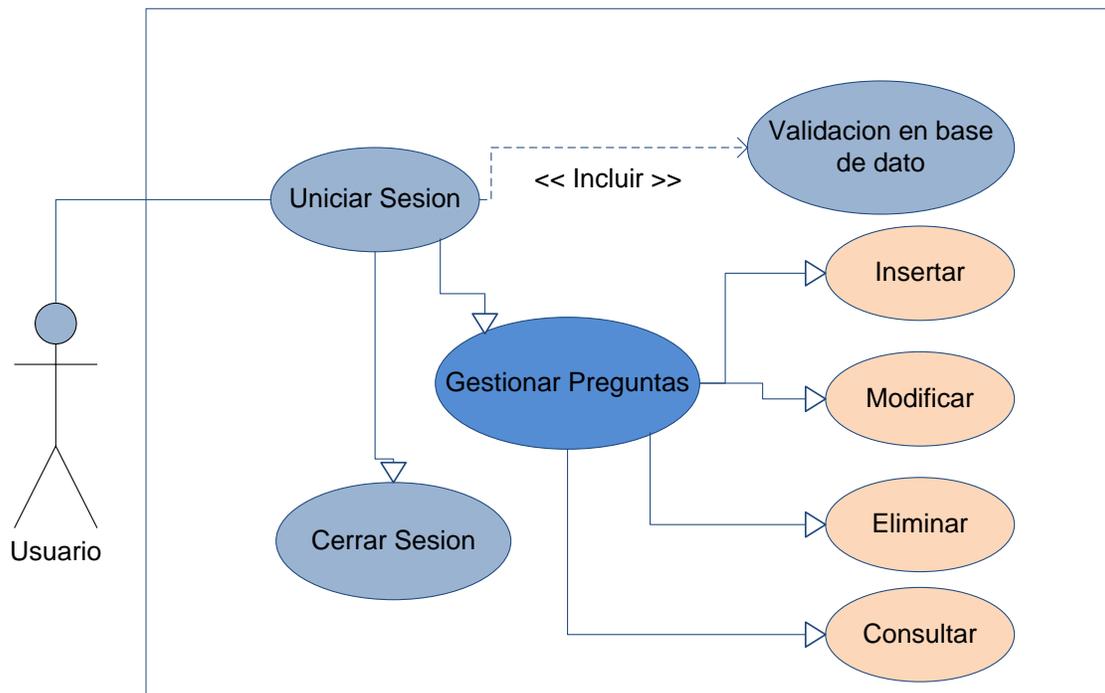


Figura 3.6. Diagrama de gestión de preguntas

Cuadro 3.5 Caso de uso Gestión de Preguntas

Caso de uso	Gestión de Preguntas	
Actor(es)	Administrador	
Descripción	Representa los procesos de insertar, modificar, eliminar y consultar preguntas de un test por parte del Administrador, previo validación de ingreso al sistema.	
Flujo	<i>EVENTOS ACTOR</i>	<i>EVENTOS SISTEMA</i>
	1. Estar autenticado en el sistema.	
	2. Estar en la sección del menú.	2. Mostrar menú de opciones
	3. Escoger la opción a realizar	3. Mostrar los elementos correspondientes a la acción seleccionada.
	4. Realiza las tareas correspondientes a la acción a realizar.	
	5. Recibir confirmación de acción exitosa.	5. Mostrar ventana de confirmación de acción.
Precondición	Estar autenticado en el sistema.	
Presunción	Sistema y servidor de base de datos disponibles.	

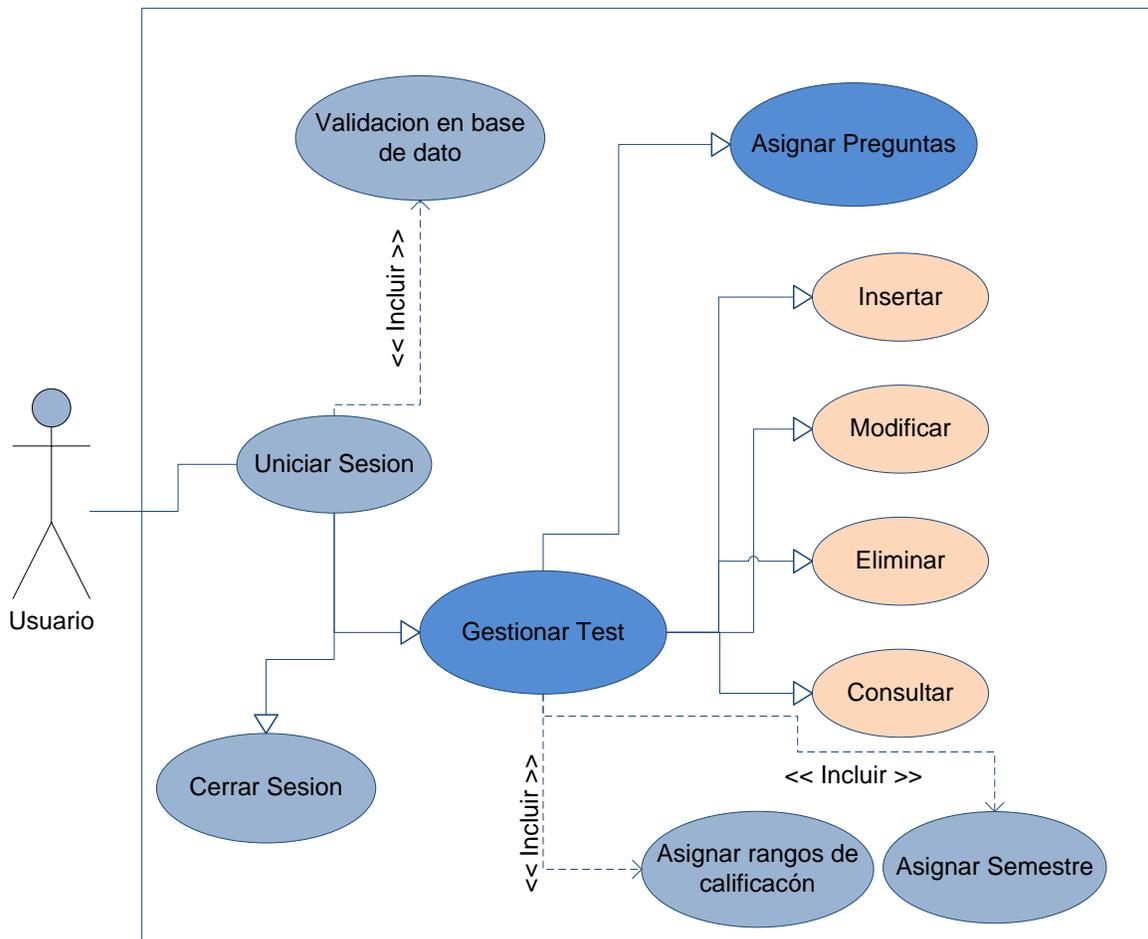


Figura 3.7. Diagrama de gestión de Test

Cuadro 3.6 Caso de uso, Gestión de Test

Caso de uso	Gestión de Test	
Actor(es)	Administrador	
Descripción	Representa los procesos de insertar, modificar, eliminar, consultar, asignar semestres, asignar rangos de calificación de Test por parte del Administrador, previo validación de ingreso al sistema.	
Flujo	<i>EVENTOS ACTOR</i>	<i>EVENTOS SISTEMA</i>

	1. Estar autenticado en el sistema.	
	2. Estar en la sección del menú.	2. Mostrar menú de opciones
	3. Escoger la opción a realizar	3. Mostrar los elementos correspondientes a la acción seleccionada.
	4. Realizar las tareas correspondientes a la acción a realizar.	
	5. Recibir confirmación de acción exitosa.	5. Mostrar ventana de confirmación de acción.
Precondición	Estar autenticado en el sistema.	
Presunción	Sistema y servidor de base de datos disponibles.	

Cuadro 3.7 Caso de uso, Asignación de preguntas

Caso de uso	Asignar Preguntas	
Actor(es)	Administrador	
Descripción	Representa el proceso de asignar preguntas a un Test previamente creado, previo validación de ingreso al sistema.	
Flujo	<i>EVENTOS ACTOR</i>	<i>EVENTOS SISTEMA</i>
	1. Estar autenticado en el sistema.	
	2. Estar en la sección de	2. Mostrar formulario de

	asignación de preguntas.	opciones.
	3. Realizar la tarea de asignación de preguntas.	
	4. Recibir confirmación de acción exitosa.	5. Mostrar ventana de confirmación de acción.
Precondición	Estar autenticado en el sistema.	
Presunción	Sistema y servidor de base de datos disponibles.	

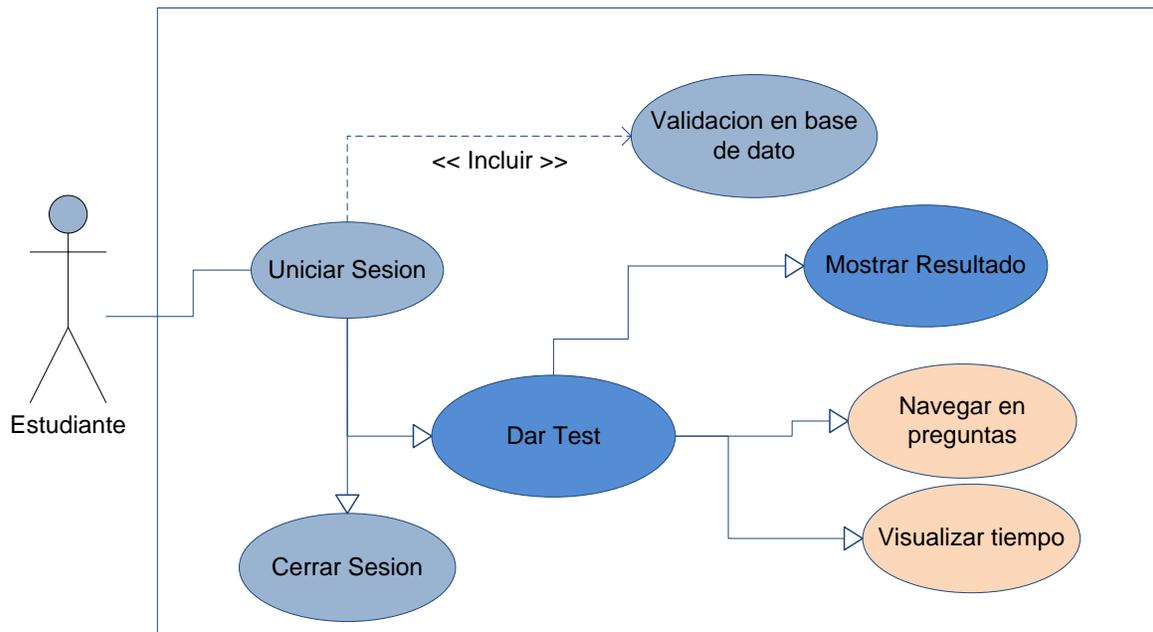


Figura 3.8 Diagrama de toma de Test

Cuadro 3.8 Caso de uso, Tomar Test

Caso de uso	Dar Test
Actor(es)	Estudiante
Descripción	Representa el proceso de dar el Test previamente creado, previo

	validación de ingreso al sistema.	
Flujo	<i>EVENTOS ACTOR</i>	<i>EVENTOS SISTEMA</i>
	1. Estar autenticado en el sistema.	
	2. Estar en la sección de dar el Test.	2. Mostrar preguntas.
	3. Realizar la tarea de contestar las preguntas.	
	4. Recibir confirmación de acción exitosa.	5. Mostrar ventana de confirmación de acción.
Precondición	Estar autenticado en el sistema.	
Presunción	Sistema y servidor de base de datos disponibles.	

Cuadro 3.9 Caso de uso, Visualización de resultados

Caso de uso	Visualizar Resultado	
Actor(es)	Estudiante	
Descripción	Representa el proceso de visualizar el resultado de un Test previamente rendido.	
Flujo	<i>EVENTOS ACTOR</i>	<i>EVENTOS SISTEMA</i>
	1. Estar autenticado en el sistema.	
	2. Estar en la sección de terminación del Test.	2. Mostrar reporte de resultados.

	3. Cerrar.	
Precondición	Estar autenticado y rendido el Test en el sistema.	
Presunción	Sistema y servidor de base de datos disponibles.	

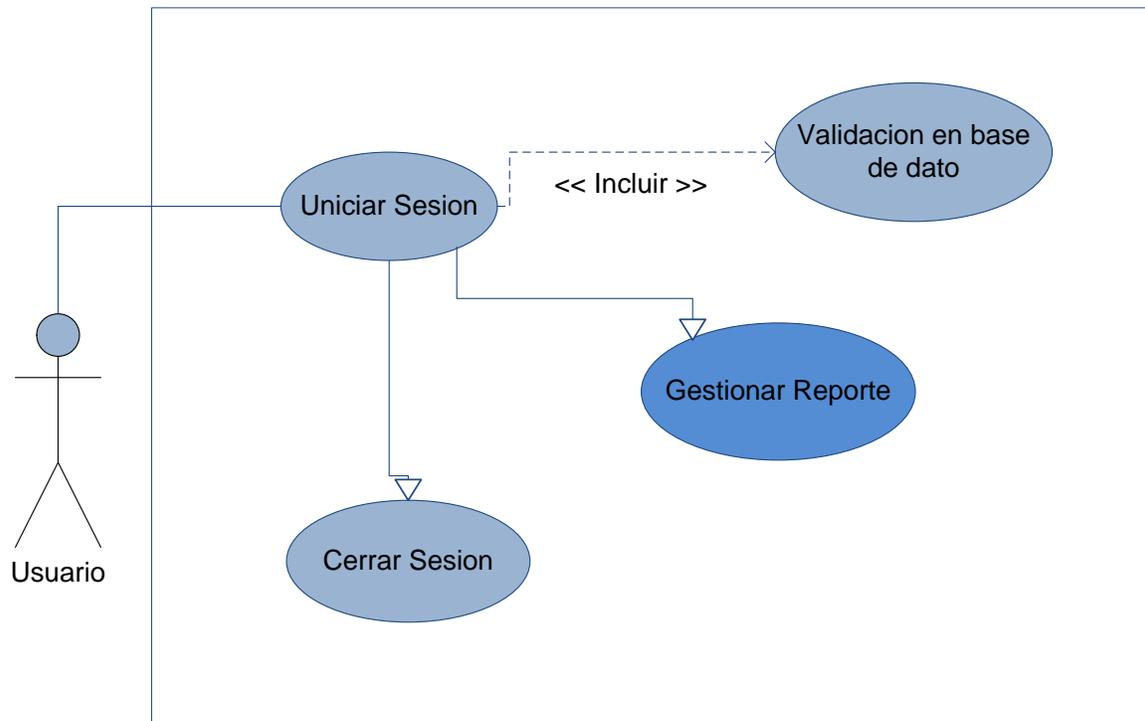


Figura 3.9 Diagrama de gestión de Reporte

Cuadro 3.20 Caso de uso, Gestionar Reportes

Caso de uso	Gestionar Reportes	
Actor(es)	Administrador	
Descripción	Representa el proceso de crear y obtener resultados estadísticos de un Test, previo validación de ingreso al sistema.	
Flujo	<i>EVENTOS ACTOR</i>	<i>EVENTOS SISTEMA</i>

	1. Estar autenticado en el sistema.	
	2. Estar en la sección de generación de reportes.	2. Mostrar formulario de opciones.
	3. Recibir reporte y confirmación de acción exitosa.	5. Mostrar reporte y ventana de confirmación de acción.
Precondición	Estar autenticado en el sistema.	
Presunción	Sistema y servidor de base de datos disponibles.	

Luego de la recopilación de requerimientos, con las consideraciones necesarias, se procedió a diseñar la base de datos del sistema, se elaboraron quince tablas que tienen privilegios para guardar, modificar y actualizar datos y doce tabla que corresponden a la base de datos y el sistema de matriculación institucional con privilegios solo de lectura. La base de datos se elaboró con el gestor de base de datos SQL-SERVER 2008 R2 como lo podemos observar en la Figura 4.10 y cuyo diccionario de datos se detalla a continuación.

Cuadro 3.11. Diccionario de datos de la tabla “Alumno”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
Id_Alumno	Int	4	N	Y	N
Fecha_Inicio	SmallDateTime	4	Y	N	N
Fecha_Egreso	SmallDateTime	4	Y	N	N
Titulo_Admission	NVarChar (50)	50	Y	N	N
Procedencia	NVarChar (150)	150	Y	N	N
Reconocimiento_Año	NVarChar (50)	50	Y	N	N
Fecha_Acta_Grado	SmallDateTime	4	Y	N	N
Nro_Acta_Grado	NVarChar (10)	10	Y	N	N
Denominacion_T	NVarChar (150)	150	Y	N	N
Fecha_Refrendacion	SmallDateTime	4	Y	N	N
Nro_Refrendacion	NVarChar (50)	50	Y	N	N

Nom_Proyecto_T	NVarChar (50)	50	Y	N	N
Id_Persona	Int	4	Y	N	Y

Cuadro 3.12. Diccionario de datos de la tabla “Asignatura”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
Cod_Asignatura	Int	4	N	Y	N
Nombre	NVarChar (150)	150	Y	N	N
Nro_Horas	Int	4	Y	N	N
Nro_Creditos	Real	4	Y	N	N
Asignatura_Previa	Int	4	Y	N	N
Cod_Carrera	Int	4	N	N	Y

Cuadro 3.13. Diccionario de datos de la tabla “Carrera”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
Cod_Carrera	Int	4	N	Y	N
Carrera	NVarChar (50)	50	Y	N	N
Seccion	NVarChar (50)	50	Y	N	N
Modalidad	NVarChar (50)	50	Y	N	N
Duracion	NVarChar (50)	50	Y	N	N
Lugar	NVarChar (50)	50	Y	N	N
Cod_Area	Int	4	N	N	N

Cuadro 3.14. Diccionario de datos de la tabla “Distribución”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
Cod_Distribucion	Int	4	N	Y	N
Cod_Semestre	Int	4	N	N	Y
Id_Docente	Int	4	N	N	Y
Cod_Asignatura	Int	4	N	N	Y
Contador	Int	4	Y	N	N
Cod_Periodo	Int	4	N	N	Y

Cuadro 3.15. Diccionario de datos de la tabla “Docente”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
Id_Docente	Int	4	N	Y	N
Titulo	NVarChar (50)	50	Y	N	N
Id_Persona	Int	4	Y	N	Y

Cuadro 3.16. Diccionario de datos de la tabla “Instrumento”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
Id_Istrumento	Int	4	N	Y	N
Cod_Matricula	Int	4	N	N	Y
IdTest	Int	4	N	N	Y

Calificacion	Real	4	Y	N	N
Aciertos	Int	4	Y	N	N
Aciertos_Pena	Int	4	Y	N	N
Errores	Int	4	Y	N	N
Equiv	VarChar (50)	50	Y	N	N
blancos	Int	4	Y	N	N
tiempoRest	Time	5	Y	N	N

Cuadro 3.17. Diccionario de datos de la tabla “Matricula”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
Cod_Matricula	Int	4	N	Y	N
Cod_Semestre	Int	4	N	N	Y
Id_Alumno	Int	4	N	N	Y
Fecha	SmallDateTime	4	Y	N	N
Estado	NVarChar (20)	20	Y	N	N
Cod_Periodo	Int	4	N	N	Y
Observaciones	NVarChar (90)	90	Y	N	N
Tipo	NVarChar (50)	50	Y	N	N
Codigo	Int	4	N	N	N
Valor	NVarChar (50)	50	Y	N	N
Cancelado	Bit	1	N	N	N
Credito	Int	4	Y	N	N

Cuadro 3.18. Diccionario de datos de la tabla “Periodo Lectivo”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
Cod_Periodo	Int	4	N	Y	N
Fecha_Inicio	SmallDateTime	4	Y	N	N
Fecha_Fin	SmallDateTime	4	Y	N	N
Nombre	NVarChar (100)	100	Y	N	N

Cuadro 3.19. Diccionario de datos de la tabla “Persona”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
Id_Persona	Int	4	N	Y	N
Nombres	NVarChar (50)	50	Y	N	N
Apellido_1	NVarChar (50)	50	Y	N	N
Apellido_2	NVarChar (50)	50	Y	N	N
Cedula	NVarChar (15)	15	Y	N	N
Genero	NVarChar (20)	20	Y	N	N
Direccion	NVarChar (50)	50	Y	N	N
Telefono_D	NVarChar (20)	20	Y	N	N
Telefono_C	NVarChar (20)	20	Y	N	N
Nacionalidad	NVarChar (50)	50	Y	N	N

Provincia_Nac	NVarChar (50)	50	Y	N	N
Canton_Nac	NVarChar (50)	50	Y	N	N
Parroquia_Nac	NVarChar (50)	50	Y	N	N
Fecha_Nac	SmallDateTime	4	Y	N	N
Email	NVarChar (50)	50	Y	N	N

Cuadro 3.20. Diccionario de datos de la tabla “Preguntas”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
IdPregunta	Int	4	N	Y	N
Pregunta	NVarCharMax	-1	Y	N	N
Img_Preg	Image	16	Y	N	N
ColumnaA	Image	16	Y	N	N
RespA	NVarCharMax	-1	Y	N	N
RespB	NVarCharMax	-1	Y	N	N
RespC	NVarCharMax	-1	Y	N	N
RespD	NVarCharMax	-1	Y	N	N
RespCorrecta	NVarChar (10)	10	Y	N	N
IdTipo	Int	4	Y	N	Y
IdPeriodo	Int	4	Y	N	N
IdTema	Int	4	Y	N	Y
nivelTaxonomico	VarChar (50)	50	Y	N	N
nivelComplejidad	VarChar (50)	50	Y	N	N
justiRespuesta	Text	16	Y	N	N

Cuadro 3.21. Diccionario de datos de la tabla intermedia “Preguntas_Test”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
IdTest	Int	4	N	Y	Y
IdPregunta	Int	4	N	Y	Y
valor	Real	4	Y	N	N

Cuadro 3.22. Diccionario de datos de la tabla “Prueba”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
IdTest	Int	4	N	Y	N
Fecha	Date	3	Y	N	N
Total_Test	Int	4	Y	N	N
Cod_Carrera	Int	4	Y	N	N
Cod_Periodo	Int	4	Y	N	N
CantidadPreguntas	Int	4	Y	N	N
TiempoPrueba	Time	5	Y	N	N
Nombre_Test	VarChar (100)	100	Y	N	N
IdTipoCalificacion	Int	4	Y	N	Y
IdTipoTest	Int	4	Y	N	Y
Estado	VarChar (50)	50	Y	N	N
horaInicio	Time	5	Y	N	N

Cuadro 3.23. Diccionario de datos de la tabla “PreguntaTest”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
IdInstrumento	Int	4	N	Y	Y
IdPregunta	Int	4	N	Y	Y
OpcionEsc	NVarCharMax	-1	Y	N	N

Cuadro 3.24. Diccionario de datos de la tabla “Rangos”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
IdRango	Int	4	N	Y	N
IdTest	Int	4	Y	N	Y
Rango_min	VarChar (3)	3	Y	N	N
Rango_max	VarChar (3)	3	Y	N	N
Descripcion	VarChar (50)	50	Y	N	N

Cuadro 3.25. Diccionario de datos de la tabla “Semestre”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
Cod_Semestre	Int	4	N	Y	N
Semestre	Int	4	Y	N	N
Paralelo	NVarChar (1)	1	Y	N	N
Cod_Carrera	Int	4	N	N	Y

Cuadro 3.26. Diccionario de datos de la tabla intermedia “Semestre_Test”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
IdSemestre	Int	4	N	Y	Y
IdTest	Int	4	N	Y	Y

Cuadro 3.27. Diccionario de datos de la tabla “Sub Área”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
IdSubArea	Int	4	N	Y	N
Nombre	NVarChar (500)	500	Y	N	N
Cod_Asignatura	Int	4	Y	N	Y

Cuadro 3.28. Diccionario de datos de la tabla intermedia “SubArea_Prueba”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
IdSubAreaPrueba	Int	4	N	Y	N
Id_SubArea	Int	4	Y	N	Y
Porc_SubArea	Int	4	Y	N	N
IdPrueba	Int	4	Y	N	Y

Cuadro 3.29. Diccionario de datos de la tabla “Tema”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
---------	--------------	--------	-------	----------------	---------------

IdTema	Int	4	N	Y	N
Tema	NVarChar (500)	500	Y	N	N
IdSubArea	Int	4	Y	N	Y

Cuadro 3.30. Diccionario de datos de la tabla “Temas Prueba”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
IdTemaPrueba	Int	4	N	Y	N
IdSubAreas_Prueba	Int	4	Y	N	Y
Porc_Temas_SubAreas_Prueba	Int	4	Y	N	N
Cantidad_Pregu	Int	4	Y	N	N
IdTema	Int	4	Y	N	Y

Cuadro 3.31. Diccionario de datos de la tabla “Tipo”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
IdTipo	Int	4	N	Y	N
Tipo	NVarChar (50)	50	Y	N	N

Cuadro 3.32. Diccionario de datos de la tabla “Tipo Calificación”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
Id	Int	4	N	Y	N
Tipo	VarChar (50)	50	Y	N	N

Cuadro 3.33. Diccionario de datos de la tabla “Tipo Test”.

Columna	Tipo de dato	Tamaño	NULOS	Clave Primaria	Clave foránea
Id	Int	4	N	Y	N
Tipo_Test	VarChar (50)	50	Y	N	N

Posterior a el diseño de la base de datos se hizo necesario probar la funcionalidad mediante la ejecución de sentencias SQL en el sistema gestor de base de datos de Microsoft SQL server 2008 R2, instrucciones que mediante sus resultados permitieron garantizar y concluir el eficiente diseño y obtener registros necesarios para la correcta funcionalidad del sistema. Entre otros se presentan las consultas que permitieron obtener los datos de estudiantes que servirán para validar que no ingresen personas para las que el test está diseñado, y obtener los resultados de las pruebas que servirá para mostrar el resultado al final de una evaluación.

SQLQuery2.sql - 1..n_Espam (sa (56))

```
-- Consulta de Estudiantes con semestre
SELECT top (100) Persona.Apellido_1 +' '+ Persona.Apellido_2 +' '+ Persona.Nombres AS NOMBRES,
Persona.Cedula AS CEDULA, Semestre.Semestre AS SEMESTRE, Periodo_Lectivo.Nombre AS PERIODO
FROM
Persona INNER JOIN
Alumno ON Persona.Id_Persona = Alumno.Id_Persona INNER JOIN
Matricula ON Alumno.Id_Alumno = Matricula.Id_Alumno INNER JOIN
Semestre ON Matricula.Cod_Semestre = Semestre.Cod_Semestre INNER JOIN
Periodo_Lectivo ON Matricula.Cod_Periodo = Periodo_Lectivo.Cod_Periodo
where Periodo_Lectivo.Nombre='SEPTIEMBRE 2013-MARZO 2014'
```

	NOMBRES	CEDULA	SEMESTRE	PERIODO
1	ESCALA LOOR GEMA DOLORES	1314793462	5	SEPTIEMBRE 2013-MARZO 2014
2	MENDOZA BRIONES MARIA BELEN	1315816890	1	SEPTIEMBRE 2013-MARZO 2014
3	ROSADO ALMEIDA GENESIS THALIA	1314994250	1	SEPTIEMBRE 2013-MARZO 2014
4	HOLGUIN BAZURTO GENESIS JAZMIN	1315185965	1	SEPTIEMBRE 2013-MARZO 2014
5	MONTEDEOCA VERA ERIKA ALEXANDRA	1313402172	6	SEPTIEMBRE 2013-MARZO 2014
6	DELGADO PALACIOS DOLORES VALENTINA	1722294921	1	SEPTIEMBRE 2013-MARZO 2014
7	MURILLO GANCHOZO GEMA MARIBEL	1313327197	6	SEPTIEMBRE 2013-MARZO 2014
8	BOMBON CUNALATA YEREMI ALDAIR	1314768142	2	SEPTIEMBRE 2013-MARZO 2014
9	BARRETO ZAMBRANO TANIA KATHERINE	1314700046	6	SEPTIEMBRE 2013-MARZO 2014
10	ZAMBRANO VALENCIA CESAR DANIEL	1314039981	5	SEPTIEMBRE 2013-MARZO 2014
11	MENDOZA VERA GRACE EDUVIG	1310773690	1	SEPTIEMBRE 2013-MARZO 2014

Figura 3.10 Consultas en SQL Server para obtener estudiantes con semestre.

SQL Query3.sql - 1..n_Espam (sa (57))

```
GO
/***** Object: StoredProcedure [dbo].[RA_Rpt_Resultados_Test] Script Date: 11/28/2013 18:16:26 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [dbo].[RA_Rpt_Resultados_Test]
@IdTest int
AS
BEGIN
SELECT Instrumento.Aciertos AS ACIERTOS, Instrumento.Errores AS ERRORES, Instrumento.blancos AS BLANCOS, LEFT(Instrumento.Equiv, 4) AS EQUIVALENTE,
Instrumento.Aciertos_Pena AS (ACIERTOS CON PENALIZACIÓN), Instrumento.Calificacion AS CALIFICACION, Prueba.Nombre_Test AS TEST,
Prueba.Fecha AS FECHA, Persona.Apellido_1 +' '+ Persona.Apellido_2 +' '+ Persona.Nombres as NOMBRES
FROM
Instrumento INNER JOIN
Prueba ON Instrumento.IdTest = Prueba.IdTest INNER JOIN
Matricula ON Instrumento.Cod_Matricula = Matricula.Cod_Matricula INNER JOIN
Alumno ON Matricula.Id_Alumno = Alumno.Id_Alumno INNER JOIN
Persona ON Alumno.Id_Persona = Persona.Id_Persona
WHERE
(Instrumento.IdTest = @IdTest)
END
GO
```

	ACIERTOS	ERRORES	BLANCOS	EQUIVALENTE	ACIERTOS CON PENALIZACIÓN	CALIFICACION	TEST	FECHA	NOMBRES
1	2	1	0	200	3	200	T-01	2013-11-12 00:00:00.000	PEREZ CALDERON ERNESTO DANIEL

Consulta ejecutada correctamente. | 186.101.70.200 (10.50 RTM) | sa (57) | Matriculacion_Espam | 00:00:00 | 1 filas

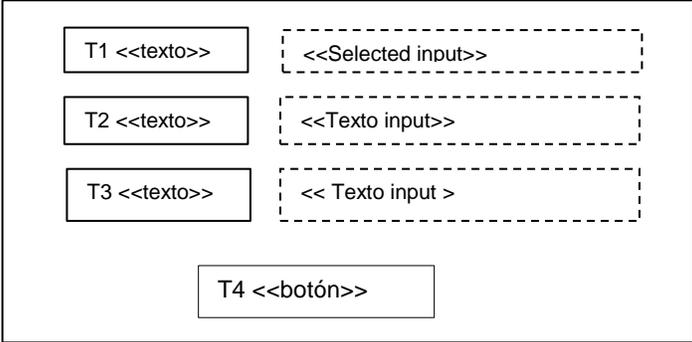
Figura 3.11 Consultas en SQL Server para obtener resultados de una evaluación.

3.2.2. FASE DE DISEÑO

Luego de cumplir con la fase de planificación, se continuó con la fase de diseño de la interfaz gráfica del sistema, esta fase consistió en la realización de los diferentes formularios web considerando los requerimientos funcionales detallados en la fase de planificación.

Se inició con la elaboración del guion técnico presentado a continuación:

Cuadro 3.34. Guion técnico del módulo Ingreso al sistema.

 <p>The diagram illustrates the layout of the login screen. It consists of a central rectangular area containing four elements: <ul style="list-style-type: none"> T1: A dropdown menu labeled '<<Selected input>>'. T2: A text input field labeled '<<Texto input>>'. T3: A text input field labeled '<< Texto input >'. T4: A button labeled 'T4 <<botón>>'. </p>
<p style="text-align: center;">TEXTO</p> <p>T1: Tipo T2: Usuario T3: Contraseña T4: ingresar</p>
<p>Descripción: Esta pantalla permite ingresar al sistema mediante un usuario y contraseña, está ubicado en el centro de la pantalla en donde se deberán seleccionar el tipo de usuario y llenar dos cajas de texto correspondiente al usuario y contraseña respectivamente, luego al dar clic en el botón “INGRESAR” y si las credenciales son correctas el usuario puede acceder al sistema con privilegios de administrador o para una evaluación.</p>

Cuadro 3.35. Guion técnico del menú de administración.

<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid black; padding: 5px; width: 15%;">T1</div> <div style="border: 1px solid black; padding: 5px; width: 15%;">T2</div> <div style="border: 1px solid black; padding: 5px; width: 15%;">Text<<input>></div> <div style="border: 1px solid black; padding: 5px; width: 15%;">Z9</div> </div>	
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; width: 10%;">Z1</div> <div style="border: 1px solid black; padding: 5px; width: 10%;">Z2</div> <div style="border: 1px solid black; padding: 5px; width: 10%;">Z3</div> <div style="border: 1px solid black; padding: 5px; width: 10%;">Z4</div> <div style="border: 1px solid black; padding: 5px; width: 10%;">Z5</div> <div style="border: 1px solid black; padding: 5px; width: 10%;">Z6</div> <div style="border: 1px solid black; padding: 5px; width: 10%;">Z7</div> <div style="border: 1px solid black; padding: 5px; width: 10%;">Z8</div> </div>	
<div style="border: 1px solid black; padding: 10px; width: 20%; margin: 0 auto;">T3</div>	
<< contenido >>	
<p>Descripción: Luego de acceder al sistema aparece una pantalla con el menú de usuario, en la parte superior aparece el nombre del sistema y en la parte izquierda la opción para cerrar la presente sesión, seguida a esta se implementó una capa que contendrá el menú de usuario. El modulo contenido es la parte del sistema que ser asignado para cargar los formularios y diferentes elementos como tablas.</p>	
<p style="text-align: center;">TEXTO</p> <p>T1: Sistema de evaluación en línea. T2: Buscar T3: Nuevo</p>	<p style="text-align: center;">ENLACES</p> <p>Z1: Sub Áreas Z2: Temas Z3: Preguntas Z4: Crear Test Z4: Diseñar Test Z6: Activar estudiantes Z7: Reportes Z8: Estadísticas Z9: Cierra la Sesión</p>

Luego de elaborar el guion técnico, se utilizó una plantilla de paneles de administración de sistemas web la cual fue modificada utilizando el Framework Bootstrap y el IDE Visual Studio 2012.

Se utilizó como base para la interfaz gráfica del sistema una plantilla estándar de paneles de administración disponible bajo licencia Public Domain como lo muestra la Figura 3.12.

CODIGO	AREA	MATERIA	OPCIONES
Fresh Web Development Resources	http://www.egrappler.com/	http://www.egrappler.com/	<input checked="" type="checkbox"/> <input type="checkbox"/>
Fresh Web Development Resources	http://www.egrappler.com/	http://www.egrappler.com/	<input checked="" type="checkbox"/> <input type="checkbox"/>
Fresh Web Development Resources	http://www.egrappler.com/	http://www.egrappler.com/	<input checked="" type="checkbox"/> <input type="checkbox"/>
Fresh Web Development Resources	http://www.egrappler.com/	http://www.egrappler.com/	<input checked="" type="checkbox"/> <input type="checkbox"/>
Fresh Web Development Resources	http://www.egrappler.com/	http://www.egrappler.com/	<input checked="" type="checkbox"/> <input type="checkbox"/>

Figura 3.12. Plantilla base del sistema

3.2.3. FASE DE CODIFICACIÓN

Inició seleccionando el modelo de programación a utilizar, se estableció la utilización de la programación bajo la arquitectura de tres capas, capa de acceso a datos, capa de negocios y capa de presentación: La capa de acceso a datos que gestiona la conexión con la base de datos diseñada en SQL Server, la capa de

negocio que gestiona la lógica del sistema, y la capa de presentación que gestionará la integración con el cliente como lo muestra la figura 4.8.

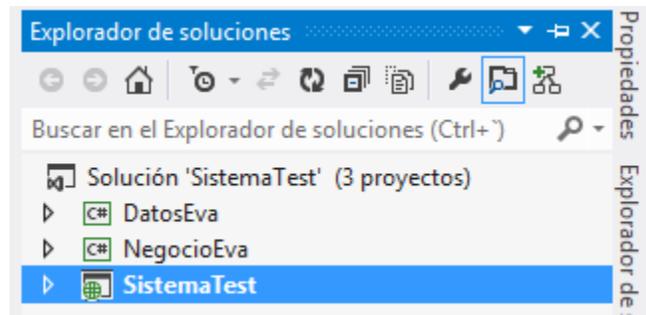


Figura 3.13. Captura de pantalla que muestra las tres capas del sistema

En la capa de acceso a datos la conexión se la gestionó utilizando Entity Framework como lo muestra la figura 4.9, para poder tratar a los elementos de la base de datos como objetos, pues, la base de datos es relacional y el lenguaje de programación es orientado a objetos, esto permitió agregar a la aplicación los procedimientos necesario para la inserción, modificación, eliminación y consultas de los datos.

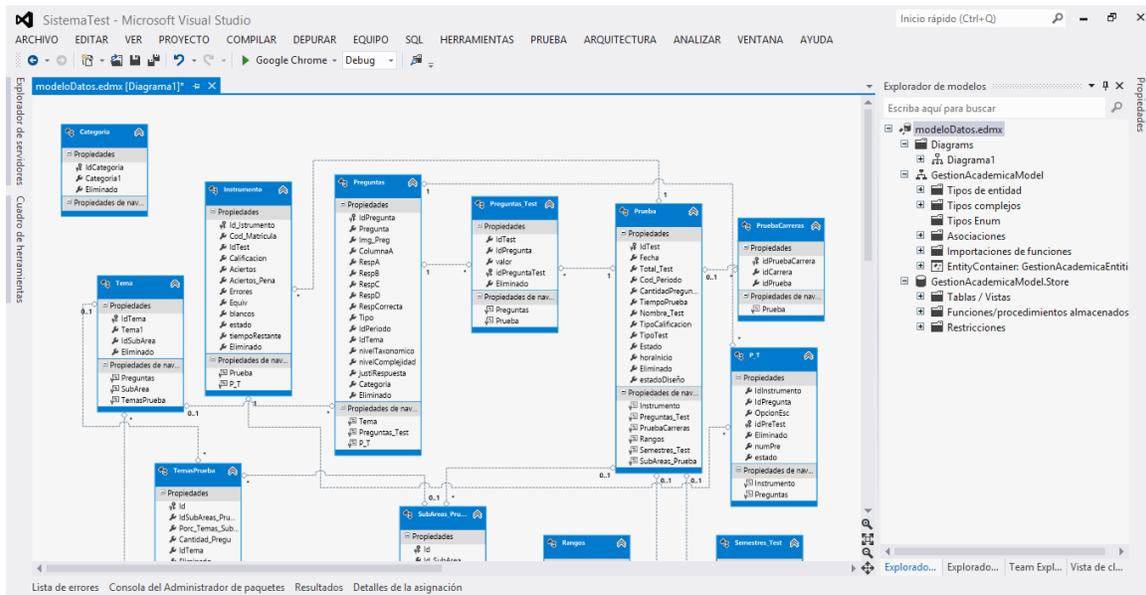


Figura 3.14. Modelo de la base de datos generado mediante Entity Framework.

En la capa de negocios se hizo uso de librerías de clases, en el lenguaje de programación C# y bajo el paradigma de la programación orientada a objetos, para construir los métodos necesarios para manipular la lógica de la aplicación, se debió considerar agregar las referencias de la capa de datos para hacer posible la utilización de los métodos que realizan la gestión con la base de datos.

A continuación se presenta el código desarrollado en la clase Prueba.

```
public class Prueba {
    public int IdTest { get; set; }
    public Nullable<System.DateTime> Fecha { get; set; }
    public Nullable<int> Total_Test { get; set; }
    public Nullable<int> Cod_Periodo { get; set; }
    public Nullable<int> CantidadPreguntas { get; set; }
    public Nullable<System.TimeSpan> TiempoPrueba { get; set; }
    public string Nombre_Test { get; set; }
    public string TipoCalificacion { get; set; }
    public string TipoTest { get; set; }
    public string Estado { get; set; }
    public Nullable<System.TimeSpan> horalInicio { get; set; }
    public string estadoDiseno { get; set; }
}
```

Figura 3.15. Atributos de la clase Prueba, Lenguaje C#

```
public Prueba() { }

public Prueba(int _idTest, DateTime? _fecha, int? _totalPrueba, int? _idPeriodo, int?
_cantidadPreguntas, string _nombrePrueba, string _TipoCalificacion, string _TipoTest, string
_estado, TimeSpan? _horalInicio, string _estadoDiseno)
{
    this.IdTest = _idTest;
    this.Fecha = _fecha;
    this.Total_Test = _totalPrueba;
    this.Cod_Periodo = _idPeriodo;
    this.CantidadPreguntas = _cantidadPreguntas;
    this.Nombre_Test = _nombrePrueba;
    this.TipoCalificacion = _TipoCalificacion;
    this.TipoTest = _TipoTest;
    this.Estado = _estado;
    this.horalInicio = _horalInicio;
    this.estadoDiseno = _estadoDiseno;
}
```

Figura 3.16. Constructores de la clase prueba.

```

public class CatalogoPrueba
{
    GestionAcademicaEntities Objeto = new GestionAcademicaEntities();

    public List<Prueba> ObtenerPrueba()    {
        List<Prueba> lista = new List<Prueba>();
        foreach (var iten in Objeto.spObtenerTest())    {
            lista.Add(new Prueba(iten.IdTest, iten.Nombre_Test,
Convert.ToDateTime(iten.Fecha), iten.TiempoPrueba, iten.Total_Test, iten.CantidadPreguntas,
iten.TipoTest, iten.TipoCalificacion, iten.Estado, iten.estadoDiseño));    }
        return lista;    }

    public string InsertarPrueba(Prueba _dato)    {
        string mensaje = "Datos guardados exitosamente";
        try
        {
            Objeto.spInsertarTest(_dato.Fecha, _dato.Total_Test, _dato.Cod_Periodo,
_dato.Nombre_Test, _dato.TipoCalificacion, _dato.TiempoPrueba, _dato.CantidadPreguntas,
_dato.TipoTest, _dato.Estado, _dato.horalnicio);
        }
        catch (Exception)
        {
            mensaje = "Error al insertar registro";
        }
        return mensaje;    }

    public string ModificarPrueba(Prueba _dato)    {
        string mensaje = "Datos modificados exitosamente";
        try
        {
            Objeto.spActualizarTest(_dato.IdTest, _dato.Fecha, _dato.Total_Test,
_dato.Cod_Periodo, _dato.Nombre_Test, _dato.TipoCalificacion, _dato.TiempoPrueba,
_dato.CantidadPreguntas, _dato.TipoTest, _dato.Estado, _dato.horalnicio);
        }
        catch (Exception)
        {
            mensaje = "Error al modificar registro";    }
        return mensaje;    }

    public string EliminarPrueba(int _datold)    {
        string mensaje = "Datos eliminados exitosamente";
        try    {
            Objeto.spEliminarTest(_datold);
        }
        catch (Exception)
        {
            mensaje = "Error al eliminar registro";
        }
        return mensaje;    }
}

```

Figura 3.17. Clase catalogoPrueba que contiene los métodos para el CRUD.

La capa de presentación es un ASP Web Form y se utilizó JavaScript, jquery, y métodos expuestos a través del objeto WebMethod, para realizar mediante la tecnología Ajax las acciones de insertar, modificar, eliminar y consultar desde un formulario web, los campos se validaron en el lado cliente mediante JavaScript y el estilo de la aplicación se lo maquetó utilizando el conjunto de herramientas disponibles en Bootstrap.

En esta última fase al ser la que tiene directa interacción con el usuario se hizo indispensable incluir las seguridades necesarias mediante validaciones de usuarios con roles diferentes (administrador y estudiante) y se utilizaron validaciones de sesiones del lado del servidor y cliente, a continuación se detalla los Métodos web utilizados.

```
[WebMethod]
public static string insertarPrueba(Prueba parDato)    {
string mensaje = "";
CatalogoPrueba _objeto = new CatalogoPrueba();
try    {
mensaje = _objeto.InsertarPrueba(parDato);    }
catch (Exception)    {
mensaje = "Error en la inserción de los datos";    }
return mensaje;    }

[WebMethod]
public static string actualizarPrueba(Prueba parDato)    {
string mensaje = "";
CatalogoPrueba _objeto = new CatalogoPrueba();
try    {
mensaje = _objeto.ModificarPrueba(parDato);    }
catch (Exception)    {
mensaje = "Error en la modificación de los datos";    }}
return mensaje;    }

[WebMethod]
public static string eliminarPrueba(int parIdPrueba)    {
string mensaje = "";
CatalogoPrueba _objeto = new CatalogoPrueba();
try    {
mensaje = _objeto.EliminarPrueba(parIdPrueba);    }
catch (Exception)    {
mensaje = "Error en la eliminacion de los datos";    }
return mensaje;    }
```

Figura 3.18. Métodos web para realizar el CRUD

Método web para la consulta de una prueba mediante filtros lambda que permiten filtrar elementos a una lista sin la necesidad de hacer una nueva consulta a la base de datos.

```
[WebMethod]
public static List<Prueba> ObtenerPruebaPorId(int parIdPrueba)  {
    CatalogoPrueba _objeto = new CatalogoPrueba();

    return _objeto.ObtenerPruebaString().Where(c => c.IdTest ==
parIdPrueba).ToList();
}
```

Figura 3.19. Método web filtrando registros mediante lambda.

Los métodos planteados anteriormente son desarrollados del lado del servidor, haciendo uso de la propiedad [WebMethod] están disponibles como servicios web. A continuación se mostrara la ejecución de estos métodos mediante Ajax desde

```
function InsertarPrueba{
    var data = { parDato: {
        Fecha: $("#txtFecha").val(),
        Total_Test: $("#txtCalificacion").val(),
        Cod_Periodo: $("#ddlPeriodo").val(),
        CantidadPreguntas: $("#txtNumPre").val(),
        TiempoPrueba: $("#txtTiempo").val(),
        Nombre_Test: $("#txtNombre").val(),
        TipoCalificacion: $("#ddlTipoCalificacion").val(),
        TipoTest: $("#ddlTipoPrueba").val(),
        Estado: $("#ddlEstado").val(),
        horalnicio: $("#TextHoralnicio").val()    }    };
    $.ajax({
        url: "frmCrudPrueba.aspx/insertarPrueba",
        data: JSON.stringify(data),
        dataType: "json",
        method: "POST",
        contentType: "application/json; charset=utf-8",
        success: function (result) {
            llenarTabla();
            alert(result.d); //Muestra el mensaje devuelto del método web
        }    });    return false;    });    }
```

Figura 3.20. Función JavaScript que ejecuta el método web de insertar.

JavaScript.

Toda la ejecución e interacción se realiza de manera más agradable y amigable con el cliente y sin realizar innecesarias recargas completa de la página e innecesarias consultas al servidor.

3.2.4. FASE DE PRUEBA E IMPLEMENTACIÓN

Una vez concluida las fases anteriores, se procedió a la implementación del sistema, con el objetivo de testear y determinar si el proyecto cumplió o no con las especificaciones consideradas al inicio del proyecto.

Para esto, se evaluó a los estudiantes de octavo, noveno y décimo de todas las carreras de la ESPAM MFL. Esta fase contempló el levantamiento de algunas funciones en el servidor de desarrollo institucional: Servidor IIS (Internet Information Services), servidores de reportes, instalación y configuración de la base de datos correspondiente al sistema, publicación y configuración del sistema web.

Las características del servidor se detallan a continuación:

- Marca: Dell.
- Modelo: PowereDge 2950 Server.
- Procesador: Intel Xeon 64bit.
- Memoria RAM: 4GB.
- Memoria Cache: 4GB.
- Fron Size bus: 2MB.
- Ray con 2 discos de 1TB SATA scsi.
- Placa base 775 2.4Ghz.
- 1 Gigabit Ethernet

Para la realización de la evaluación, fue necesario ingresar en el sistema las áreas, temas y preguntas relacionadas a la misma. Luego se ingresó y configuró la evaluación a realizar, es decir ingresar al sistema los semestres que se evaluarán, y los rangos de calificación, en esta parte también se determinaron los porcentajes y valores correspondientes a cada área, tema y pregunta.

La implementación se la realizó en el laboratorio 201 del edificio de la carrera de informática de la ESPAM MFL, aquí se activaron 26 computadoras que se necesitarían para cubrir la demanda de cada semestre. Se hizo necesario configurar las ventanas emergentes del navegador Google Chrome, ya que este navegador brinda las características necesarias por parte del sistema. Se destaca que el sistema está optimizado para este browser.

Se evaluaron 396 personas desde el 21 de enero al 4 de febrero del 2014 correspondiente a octavo, noveno y décimo semestre de las carreras de medio ambiente, informática, agroindustrias, pecuaria, agrícola, turismo, administración de empresas públicas y privadas.

La evaluación tuvo una duración de 60 minutos por semestre, 70 preguntas relacionadas a las áreas de comprensión lectora, y comprensión escrita.

CAPÍTULO IV. RESULTADOS Y DISCUSIÓN

El autor luego de realizar todas las actividades necesarias y consideradas según la ingeniería de requerimientos y haciendo la recolección, análisis y estructura de la información, se obtiene como resultado la base de datos modelada en SQL SERVER 2008 R2 como sistema gestor de base de datos.

Estas actividades permitieron obtener una base de datos con 15 tablas principales, 7 tablas relacionales y 12 tablas existentes en la base de datos institucional del sistema de matriculación, que contiene información sobre asignaturas, distribución, docentes, estudiantes, periodos, materias, carreras y semestres, mismas que permitieron a través de consultas obtener datos requeridos para el sistema informático de test on-line, por este motivo se hizo indispensable trabajar en coordinación con la unidad de producción de software, unidad que administra la base de datos y el sistema de matriculación institucional, así se obtuvo la base de datos que muestra la figura 4.1. que sirvió como plataforma para la continuidad del desarrollo del sistema.

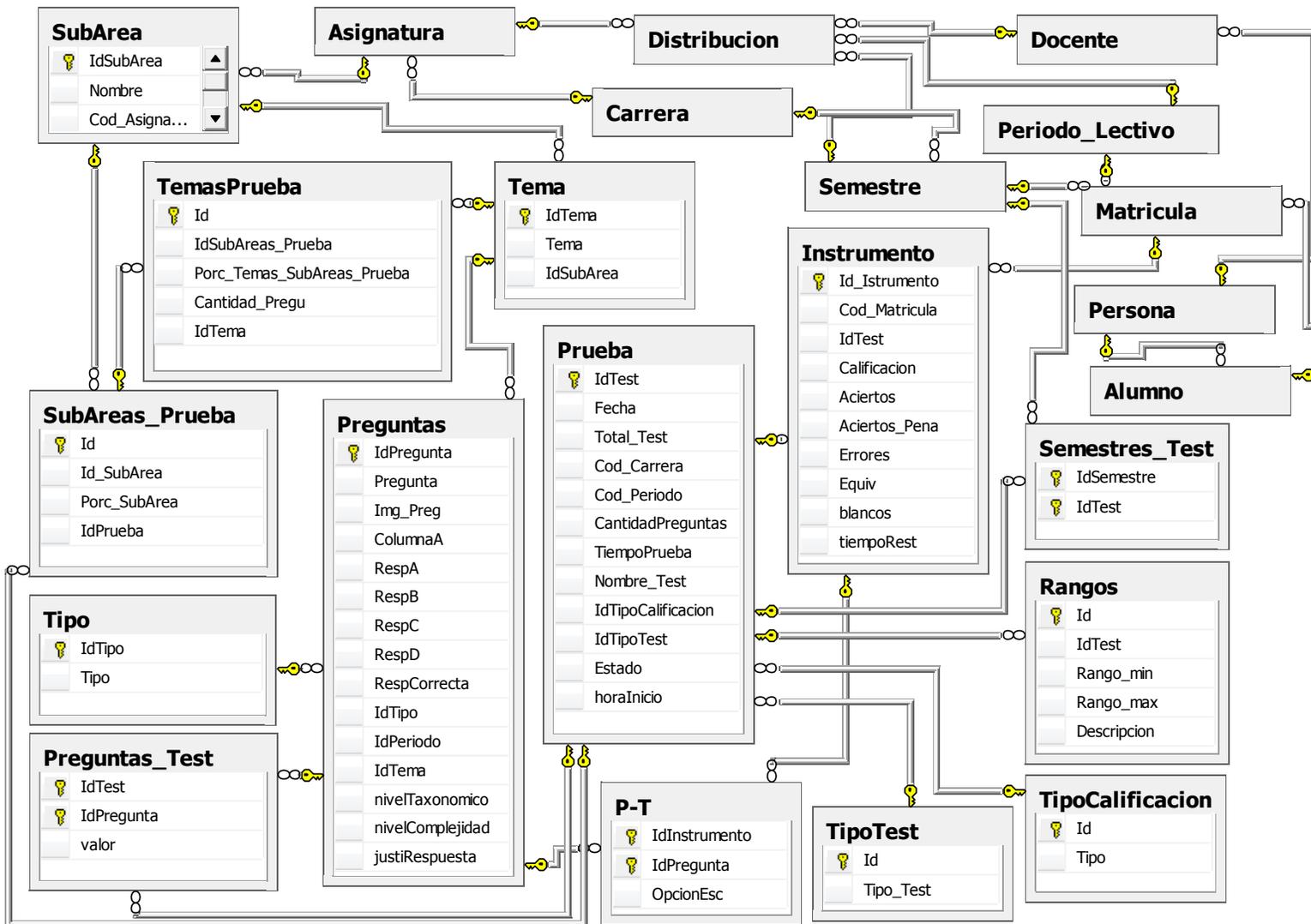


Figura 4.1 Diagrama de base de datos en SQL SERVER 2008 R2

Terminado el análisis de los requerimientos y definida la estructura de la base de datos se procedió a la elaboración de la interfaz de usuario del sistema, misma que fue diseñada tomando como base una plantilla de administración libre, y Bootstrap como conjunto de herramientas de desarrollo web, para lograr obtener un diseño intuitivo, de fácil interacción y de diseño responsivo para que se adapte a diferentes medidas de pantalla.

Terminado el diseño de las interfaces gráficas, se hizo uso de la entrevista como técnica, para evaluar que la interfaz resultado cumpla las características necesarias por parte del comité de evaluación institucional y estudiantes. A continuación se presentan los resultados obtenidos que permiten concluir que la interfaz fue aprobada satisfactoriamente.

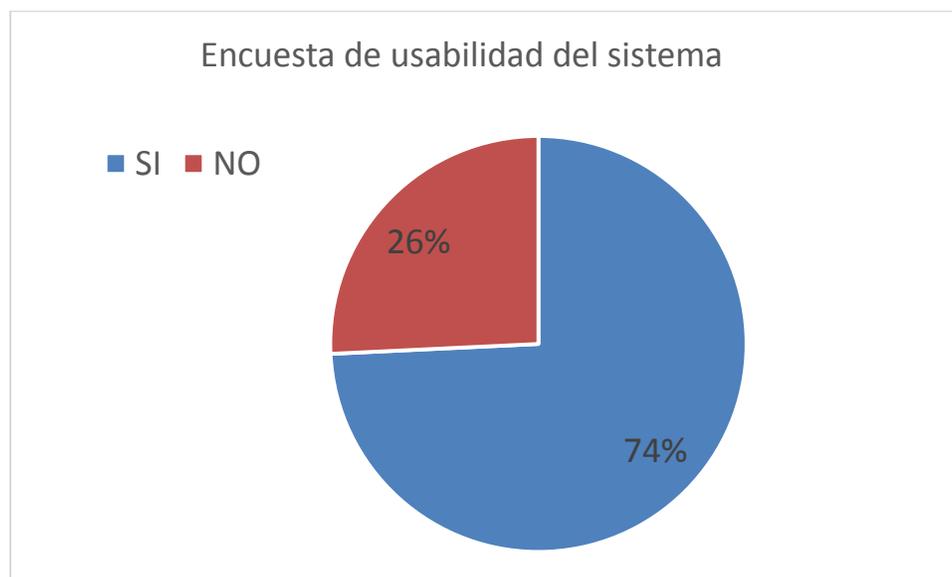


Gráfico 4.1 Resultados entrevista de valoración de interfaces.

Al tabular los datos obtenidos en relación a la interfaz gráfica del sistema, considerando una muestra de 66 personas se concluye que en un 74% de los preguntados consideran la interfaz del sistema funcional, agradable, fácil y de intuitiva interacción.

Luego del proceso de presentación y aprobación de las interfaces de usuario, se presentan y describen algunas.



The image shows a login form titled "Sistema de Gestion de Evaluaciones en linea | ESPAM MFL". Below the title is the subtitle "Credenciales de usuario". The form contains three input fields: a dropdown menu with "ADMINISTRADOR" selected, a text field with "adrian" and a user icon, and a password field with "....." and a key icon. A green "Validar" button is located at the bottom right of the form.

Figura 4.2 Formulario de validación de usuarios.

La figura 4.2. presenta el módulo de seguridad del sistema, formulario que permite mediante credenciales de usuarios el ingreso al sistema con dos roles diferentes, administrador y evaluación, para el ingreso al Backend (menú de administración) y al frontend (módulo de evaluación) respectivamente.

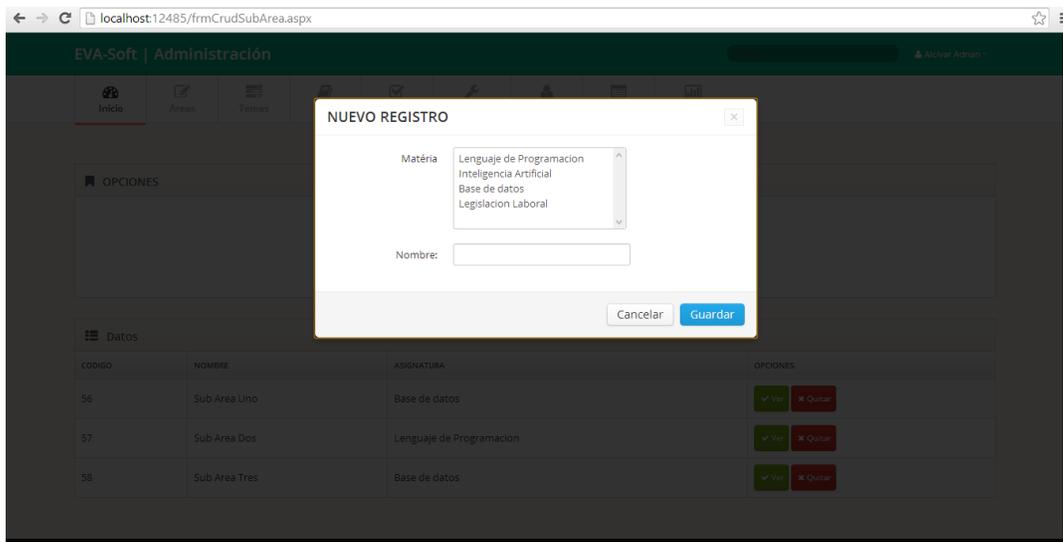


Figura 4.3. Pantalla de gestión de Áreas

La figura 4.3. muestra la pantalla que permite el ingreso, modificación y eliminación de las Áreas asignadas a una asignatura, mediante JQuery y javascript se utilizan ventanas modales para generar una experiencia más intuitiva y participativa con el usuario.

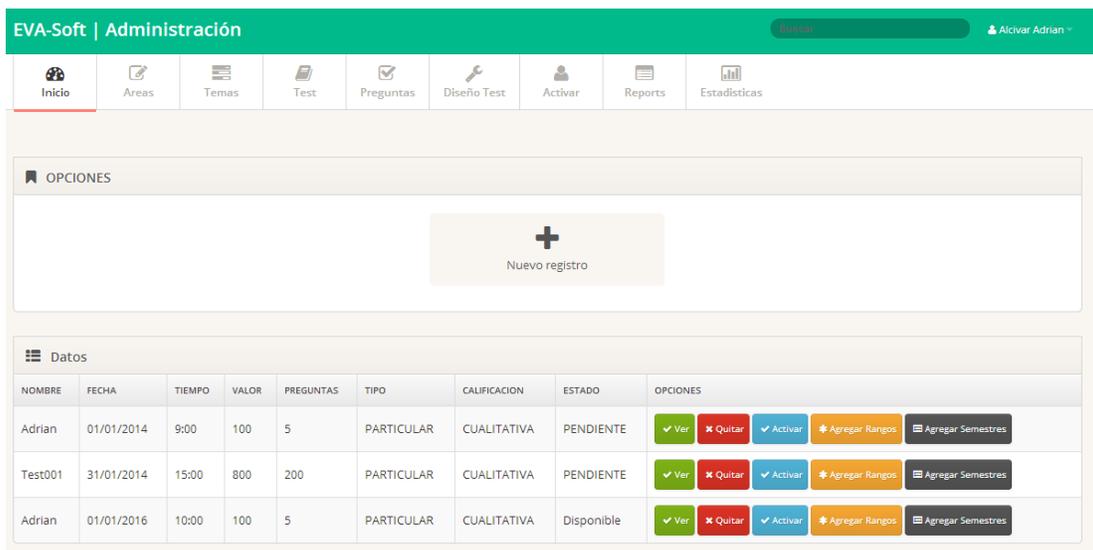


Figura 4.4 Pantalla de gestión de Evaluaciones

La figura 4.4 muestra el formulario web que permite la gestión de las evaluaciones, admite las operaciones de crear, modificar, eliminar y consultar evaluaciones,

además asignar las carreras participantes en la evaluación, semestres y rangos de calificación entre otras operaciones como activar una evaluación.

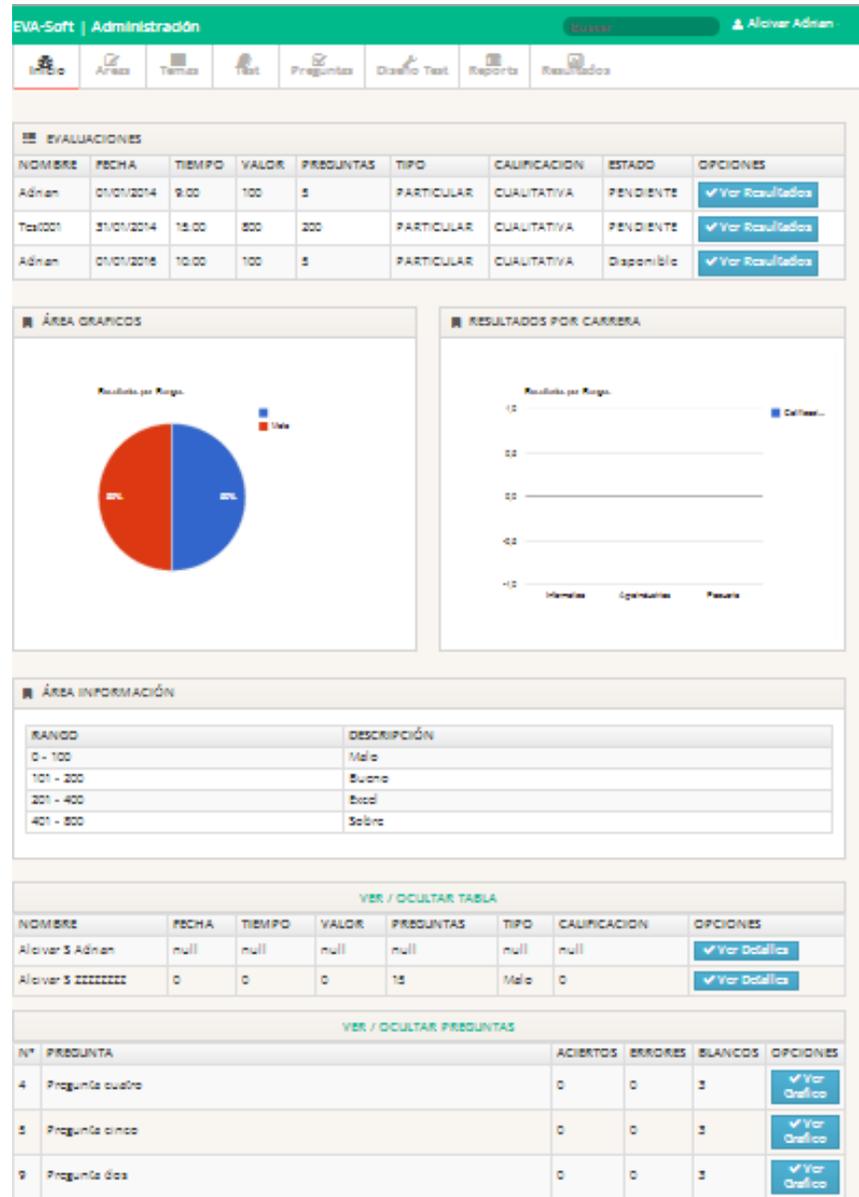


Figura 4.5. Formulario de reportes web

Posterior al diseño de las interfaces se procedió a la programación de la lógica de aplicación, fase de la cual resultaron los diagramas de clases correspondientes a las diferentes capas del sistema que se muestran seguidamente.

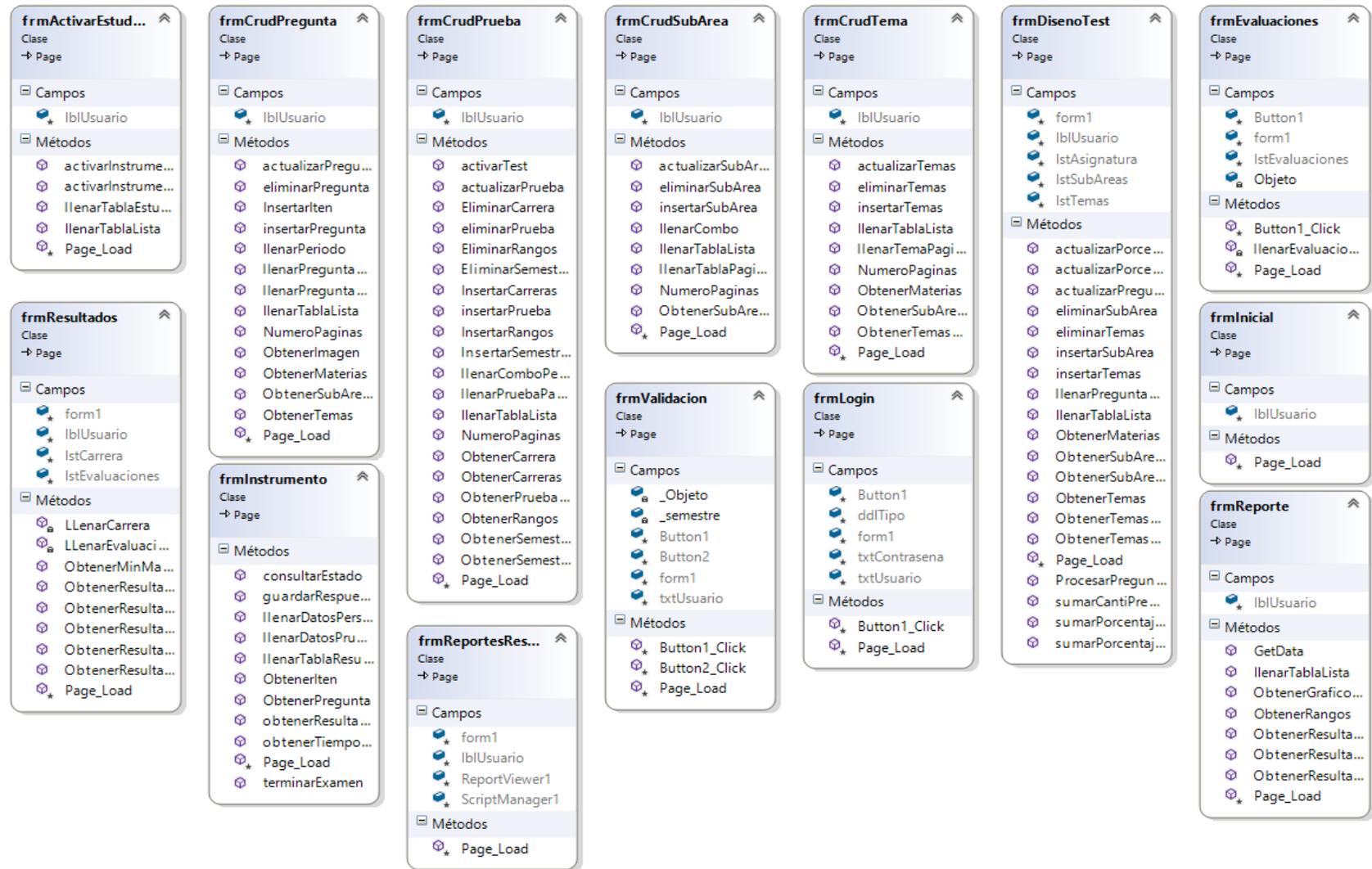


Figura 4.6. Diagrama de clases de la capa presentación

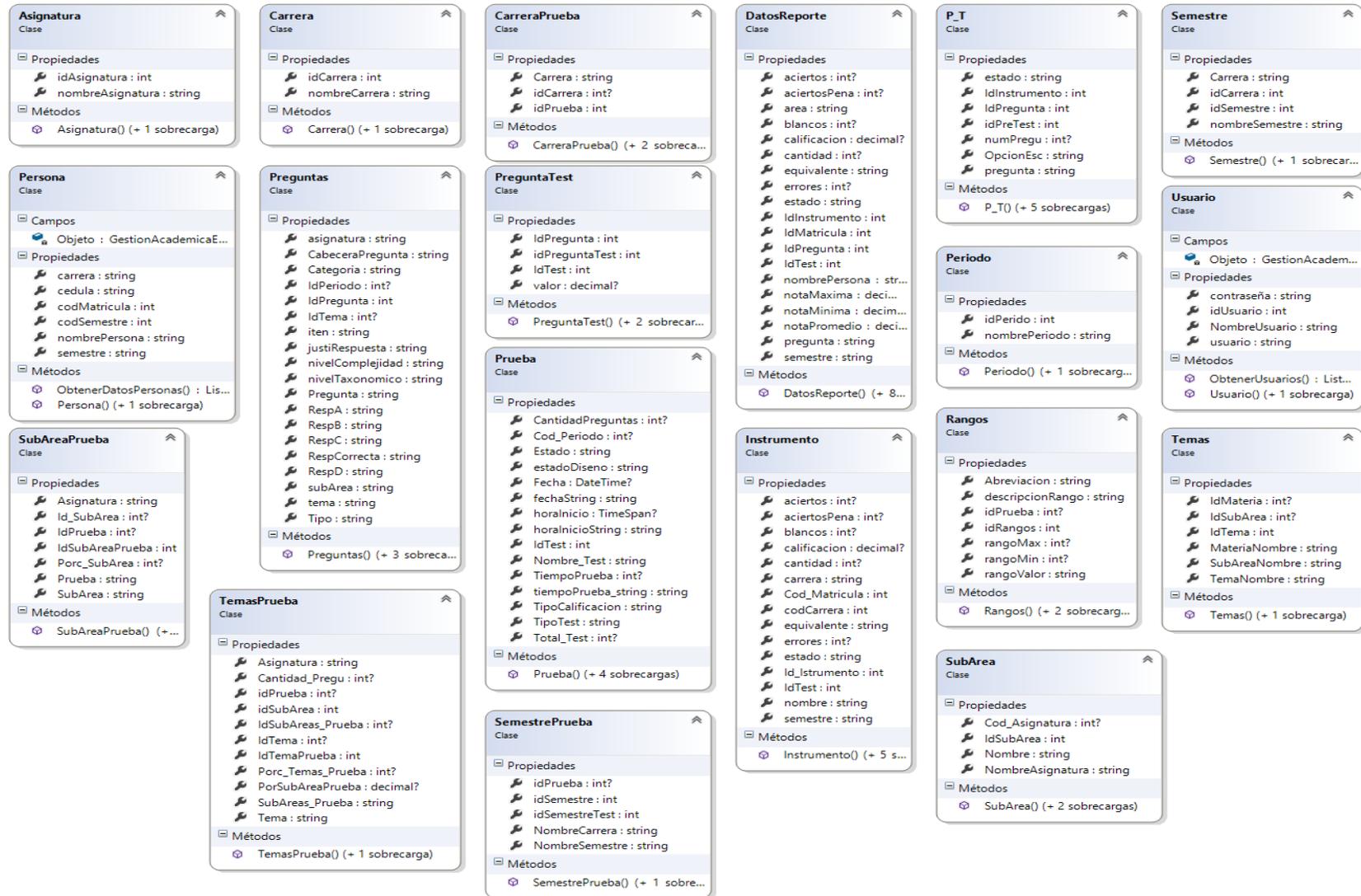


Figura 4.7. Diagrama de clases tontas de la capa negocio

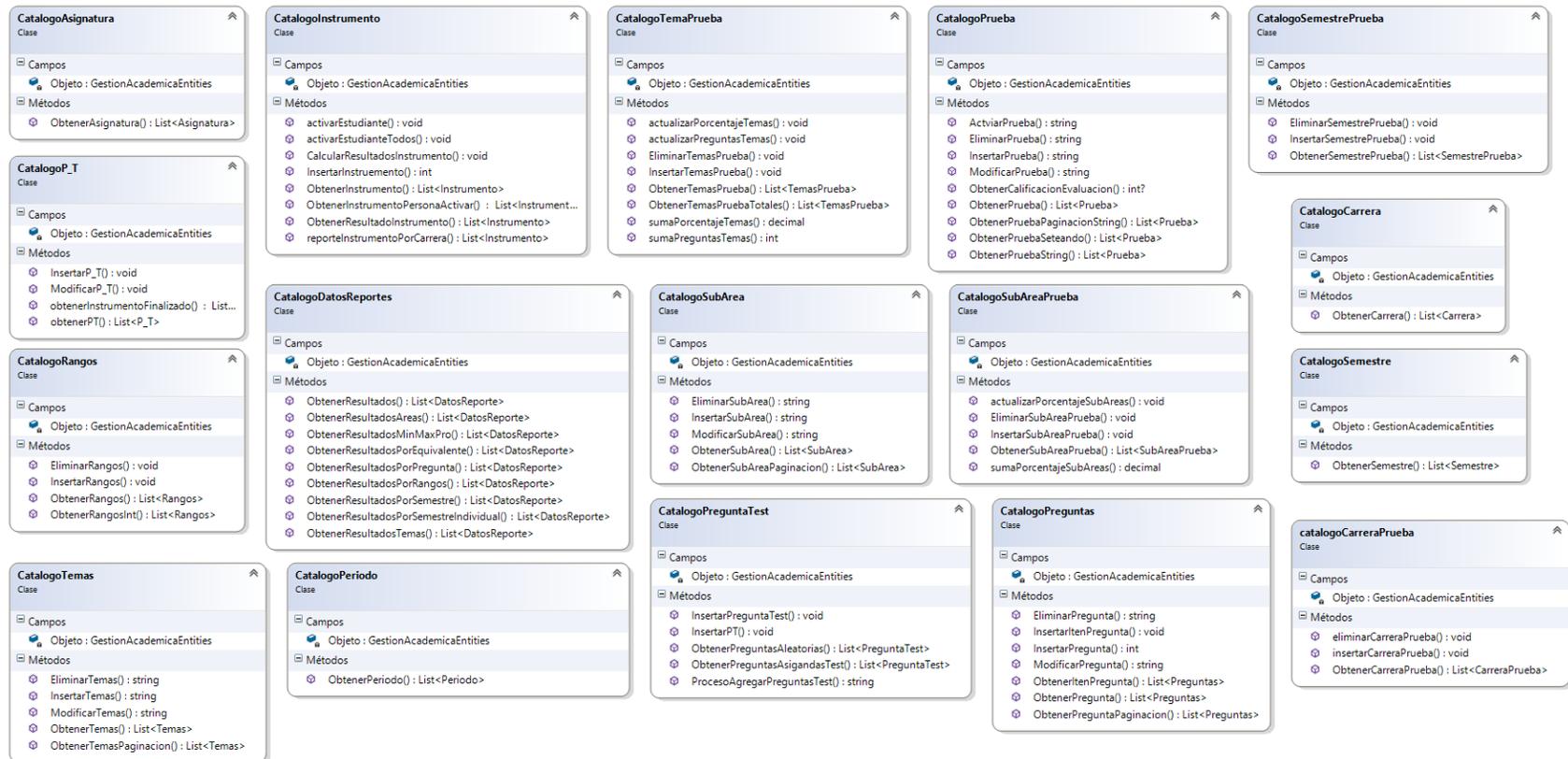


Figura 4.8. Diagrama de clases de catálogos de la capa negocio



Figura 4.9. Diagrama de clases de la capa de datos

The screenshot shows a web interface for an evaluation. At the top left is the logo of the Universidad Tecnológica de Heracleito (UTM). To the right, student information is displayed: Carrera: INFORMÁTICA, Semestre: 10 A, Cedula: 1312499625, Código Evaluación: 01 COMPRENSIÓN LECTURA, Estudiante: ALCIVAR SANTANDER ADRIAN HERACLITO, and Tiempo: 60. Below this, a green bar indicates 'Pregunta 8 de 70' with buttons for 'Mostrar tiempo' and 'Terminar'. The main question asks to identify the correctly accented word among four options: A. Cástor, B. Vendería, C. Hiáto, and D. Ilegái. Navigation buttons for 'Anterior' and 'Siguiente' are at the bottom.

Figura 4.10. Pantalla web de evaluación

La figura 4.10 muestra la pantalla que muestra y brinda las opciones de respuestas a los estudiantes, muestra información del evaluado e información del test. Al concluir mostrara la información resultante de la evaluación, es decir la calificación, cantidad de aciertos, errores, blancos y otros datos necesarios como lo muestra la figura 4.11.

The screenshot shows the results page. At the top, the same student information as in Figure 4.10 is displayed. Below it, the heading 'RESULTADOS' is followed by a table with the following data:

INFORMACIÓN	
CALIFICACION	124.66
EQUIVALENTE	INSATISFACTORIO
ACIERTOS	12
ACIERTOS PENALIZACION	0
ERRORES	41
BLANCOS	17

Figura 4.11. Tabla resultante al terminar la evaluación.

Concluida la evaluación el sistema, después de evaluar a los estudiantes de octavo, noveno y décimo semestre de todas las carreras, se presentan algunas evidencias de esta fase.

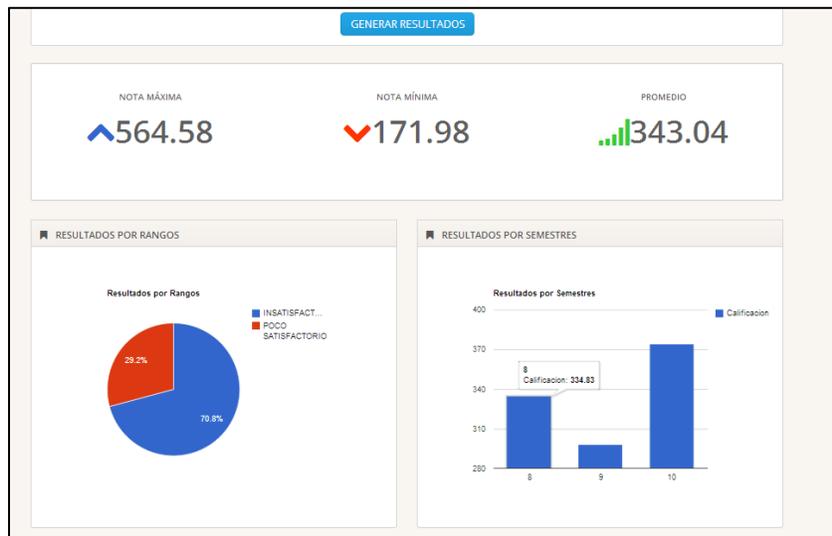


Foto 4.12. Resultados evidenciados segundos después de terminar de evaluar

En conversaciones con miembros del comité de acreditación institucional se evidencio la optimización en el proceso de evaluación de los estudiantes, a continuación se muestra un cuadro comparativo.

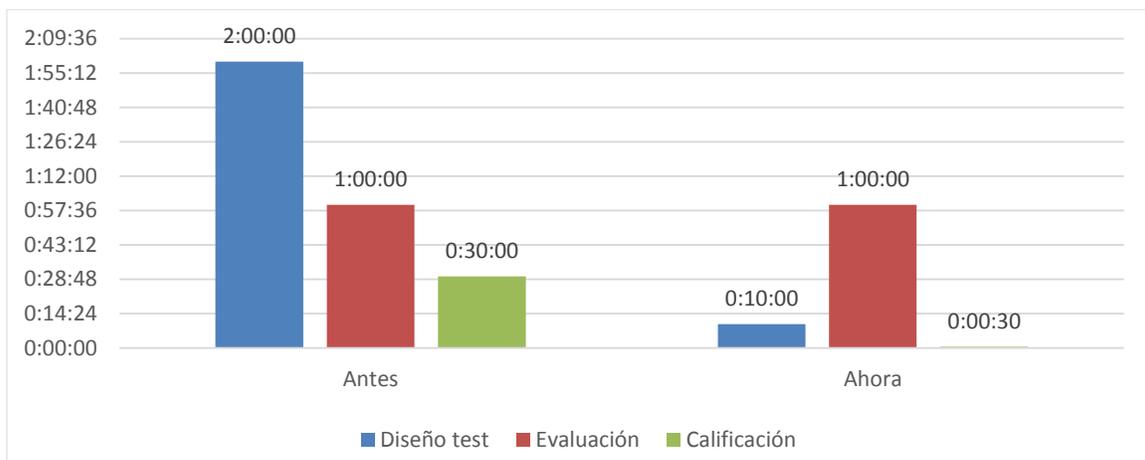


Grafico 4.1. Grafico comparativo de tiempos de diseño, evaluación y calificación de un test.

El grafico 4.1 presenta datos relacionados al tiempo que necesitaba la coordinación de evaluación en diseñar una evaluación, sin el sistema se invertía al menos dos horas, mediante el sistema el tiempo de diseño no excedió los 10 minutos en el mayor de los tiempos, de igual manera el tiempo de calificación individual de cada evaluación era de más de 20 minutos en un test de 70 preguntas, con el sistema, este proceso se automatiza y se pueden evidenciar los resultados posterior a los 30 segundos terminada la evaluación.

En esta fase se hizo necesario cuantificar el tiempo de respuesta del sistema, para esto se utilizó dos herramientas disponibles en: <http://developers.google.com/speed/pagespeed/insights> herramienta de Google para desarrolladores, y <http://tools.pingdom.com/fpt>, de las cuales se obtuvo un grado de rendimiento de 87/100 y 88/100 como lo evidencian las figuras 4.13 y 4.14 respectivamente.



Figura 4.13. Prueba de rendimiento del sistema desde tolos.pingdom.com.

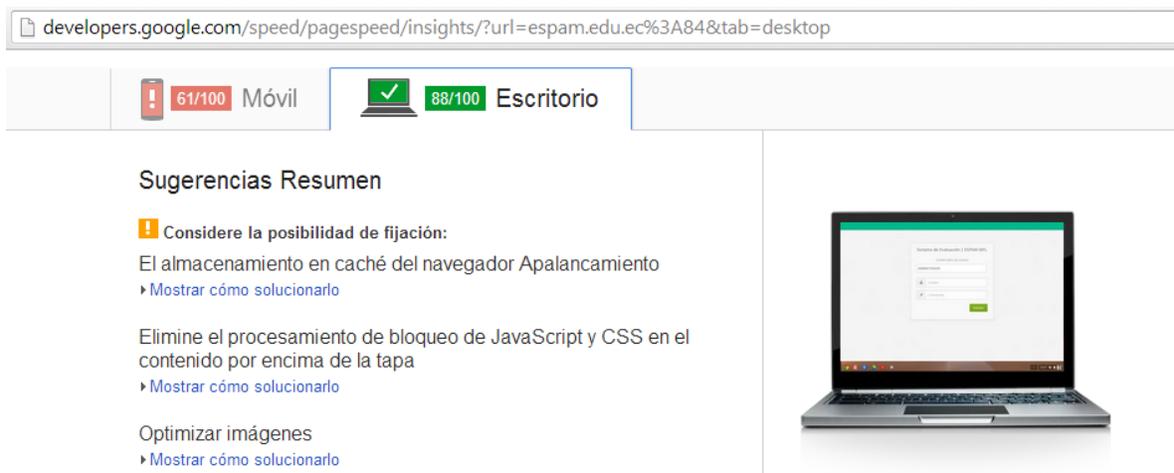


Figura 4.14. Foto de prueba de rendimiento del sistema desde <http://developers.google.com/speed/pagespeed/insights>

ANÁLISIS COSTO BENEFICIO

El presente cuadro comparativo pretende hacer un análisis comparativo del costo del desarrollo del sistema de evaluación en línea real y el costo proyectado en el mercado.

Cuadro 4.1. Datos relacionados al análisis costo beneficio.

ROLES	COSTO REAL	COSTO MERCADO
Gestión Proyecto		
• Proyect Manager	\$0.00	\$1600.00
• Equipo cliente	\$0.00	\$0.00
• Asesores Externos	\$0.00	\$200.00
Software		
• Scrum manager		
• Team development	\$0.00	\$1400.00
• Team test		
• Team design	\$1280.00	\$1280.00
Capacitación	\$0.00	\$1280.00
• Team	\$0.00	\$1280.00
Implementación		
• Team	\$0.00	\$200.00
Soporte		
• Team	\$0.00	\$1280.00
	\$0.00	\$1280.00
OTROS		
Equipos		
• Hosting	\$0.00	\$200.00
• Computadoras		

<ul style="list-style-type: none"> • Estructura de red 	\$1200.00	\$1200.00
Logística	\$0.00	\$400.00
<ul style="list-style-type: none"> • Transporte • Alimentación 	\$135.00	\$135.00
Otros	\$75.00	\$75.00
<ul style="list-style-type: none"> • Papelería 	\$20.00	\$20.00
TOTAL	\$2710.00	\$10550.00

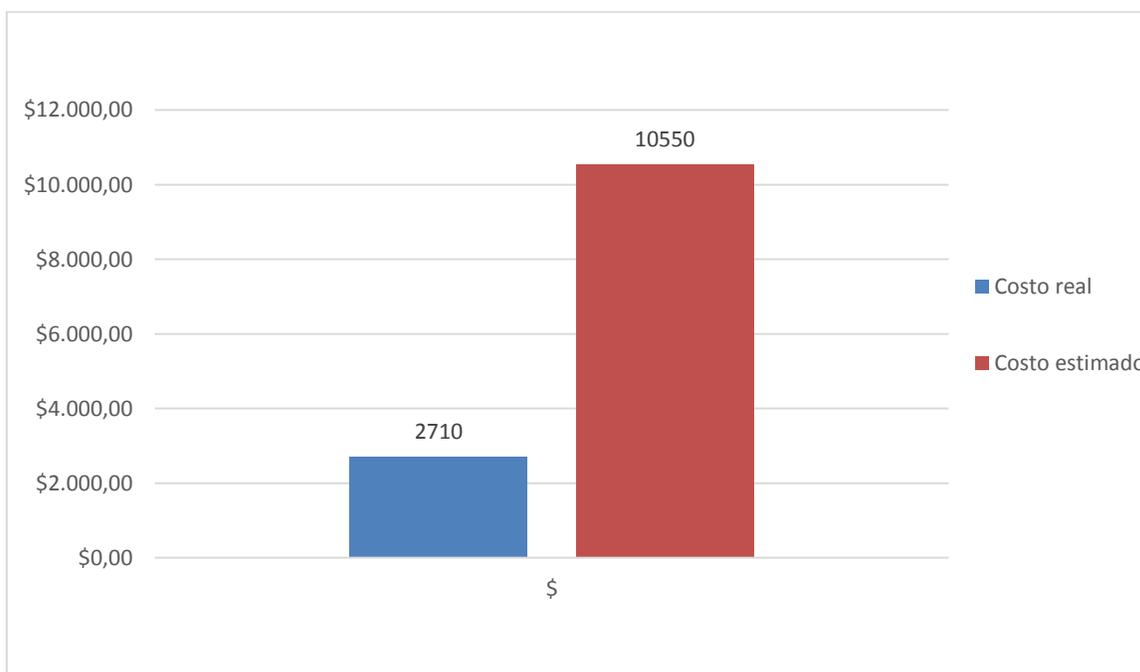


Grafico 4.2. Grafico comparativo costo real, esperado

El grafico 4.2. Detalla el costo que significaría para la ESPAM MFL desarrollar el sistema de evaluación en línea ascendiendo a \$10550.00 dólares y hace una comparación con el costo real al desarrollarlo como proyecto de tesis y haciendo uso de los equipos existentes en la universidad, entre los cuales podemos

mencionar los servidores, la estructura de red, el Project Manager (Ing. Ángel Vélez), el Scrum Manager (Ing. Jessica Morales) y otros costos.

DISCUSIÓN

Citando el trabajo de investigación titulado sistema de generación de tests on-line para dispositivos móviles, que agrupa varios procesos que facilitan la creación, administración, evaluación y realización de exámenes tipo test (preguntas respuesta corta, múltiple opción y emparejamiento). Está dividido en cuatro módulos. Generador de preguntas: Generador de Cuestionarios: Visor de Exámenes: Administrador de Test (Céspedes y González. 2012).

Tomando como referencia el sistema mencionado en el agregado anterior, el autor desarrolló el sistema de generación de test online para el comité de evaluación de la ESPAM MFL con la finalidad de optimizar el proceso continuo de evaluación de los estudiantes. Se lo desarrolló implementando funcionalidades extras al trabajo de investigación mencionado anteriormente, así, es posible mencionar la característica de diseño de un test en base a parámetros ingresados y asignando las preguntas aleatoriamente, muestra a más de reportes en documentos planos, servicios que permiten mostrar dinámicamente resultados directamente sobre el navegador. Información gráfica y estadística de una evaluación, asignar semestres con la finalidad de impedir que estudiantes no autorizados rindan exámenes, configuración de forma de mostrar el resultado y los rangos asignados pueden configurarse para cada evaluación, tiene más opciones de configuración y personalización de un test, crear preguntas y asignarlas a alguna materia con la finalidad de tener una base de datos de reactivos, asignar preguntas aleatorias a un examen, cambiar preguntas en caso de ser necesario y el estudiante en el módulo de rendir el examen podrá marcar preguntas con la finalidad de al final del test revisarlas en caso de dudas.

Al final el sistema mostrara información con la puntuación obtenida y guardara la información correspondiente en la base de datos.

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

La fase de análisis de requerimientos permitió realizar el levantamiento de la información necesaria, que posteriormente se tradujo en requerimientos funcionales y en una base de datos estructurada y robusta.

La implementación de Bootstrap como Framework de desarrollo de interfaces web proporciono el conjunto de herramientas necesarias para obtener el diseño de interfaces agradable y facilitó la interacción del usuario con el sistema.

El desarrollo del sistema mediante la metodología en tres capas y bajo el paradigma de programación orientada a objetos permitió organizar y estructurar el código eficientemente, y facilitará futuros soportes y cambios al sistema.

La implementación de un sistema de evaluación en línea permitió evaluar y mostrar a los estudiantes de la ESPAM los resultados al finalizar la evaluación y más eficientemente.

5.2. RECOMENDACIONES

Realizar un registro y análisis exhaustivo de los requerimientos expresados en lenguaje natural en las reuniones iniciales y validar que la traducción a requerimientos finales sea la necesaria para que el sistema cumpla las expectativas planteadas.

Investigar e implementar en los proyectos de investigación de la universidad marcos de trabajo existentes con la finalidad de interactuar con tecnologías actuales que facilitaran y ahorran tiempo en el proceso de desarrollo del sistema.

Usar el sistema no solo por parte del comité de investigación sino los docentes en general, y así contar con una herramienta que les automatice el proceso de evaluar.

Desarrollar funcionalidades extras para permitir configurar evaluaciones de práctica, con la finalidad de mantener bancos de preguntas actualizadas y disponibles para todos los estudiantes interesados en medir sus competencias.

BIBLIOGRAFÍA

- Abadal, E. 2001. Sistemas y servicios de información digital. España. Asturias. Edicions Universitat Barcelona. Vol 47. p 79.
- Acosta, J; Greiner, C; Dapozo, G; Estayno, M; 2012. Medición de atributos POO en frameworks de desarrollo PHP. Departamento de Informática. Facultad de Ciencias Exactas y Naturales y Agrimensura Universidad Nacional del Nordeste. Revista Universidad. ARG. Vol. 1. p 1-10.
- Alvarez, E y Montes, M. 2010. Sistema de gestión de requisitos para titulaciones en la Universidad de Cuenca.
- Anaya, R. 2012. Una visión de la enseñanza de la Ingeniería de Software como apoyo al mejoramiento de las empresas de software. Revista Universidad EAFIT. vol 42, p 60-76.
- Avila, E y Meneses, A. 2013. Delfdroid y su comparación evaluativa con XP y Scrum mediante el método 4-DAT. Revista Cubana De Ciencias Informáticas. Vol. 7. p 16-23.
- Cabero, J. 2006. Bases pedagógicas del e-Learning. España. Revista de universidad y sociedad del conocimiento. 1 ed. Vol 3. p 1-10.
- Céspedes, E y González, M. 2012. Desarrollo e implementación de un sistema de generación de tests on-line para dispositivos móviles. p 1-8.
- Chaves, J; Medina, F; Arbeláez, O. 2011. Herramientas para el desarrollo rápido de aplicaciones web. Scientia Et Technica. vol 1, p 254 - 258.
- Daniele, M; Uva, M; Martelloto, P; Picco, G. 2010. Aplicación de herramientas CASE a la enseñanza de Ingeniería de Software: Gestión de la Configuración de Software y Testing Funcional. Universidad Nacional de Río Cuarto. Cordoba. ARG. p 1-9.
- De la Torre, A. 2006. Web Educativa 2.0. Revista electrónica de tecnología educativa. Vol. 20. p 1-10.

- Escobar, J; Losavio, F; Ortega, D; 2012. Una revisión de Frameworks, Lenguajes de Modelado y Herramientas para Arquitecturas Empresariales.
- Escontrela, R y Stojanovic, L. 2004. La integración de las TIC en la educación: apuntes para un modelo pedagógico pertinente. Colombia. Rev. Pedagógica. Vol 25. p 74
- Falappa, M; Cobo, M; Martínez, D; Benedetto, M; Carabio, A; Alvez, C; Cabrera, S; 2013. Estudio comparativo y análisis de rendimiento de los lenguajes de manipulación de datos en bases de datos orientadas a objetos y bases de datos objeto-relacionales. XV Workshop de Investigadores en Ciencias de la Computación.
- Fernández, A. 2010. La evaluación orientada al aprendizaje en un modelo de formación por competencias en la educación universitaria. Norteamérica. REDU. Revista de Docencia Universitaria.
- García, A. 2011. Integración de las TIC en la docencia Universitaria. 1 ed. España. La Coruña. p 81.
- Guevara, R. 2008. Sentencias básicas usadas en la programación de computadores. 1 ed. Medellín-Colombia. p 15.
- Haughee, E. 2013. Instant Sublime Text Starter. Packt Publishing Ltd. En línea. Disponible en: <http://goo.gl/hCYO4A>
- Hechavarría, D y Herrera, L. 2012. Procedimientos para la gestión de requisitos en el SCADA SAINUX. Revista Cubana De Ciencias informáticas, Vol. 6. N 4. P 27-33.
- Heredia, J; Álvarezza, L; Linares, N; 2011. Comparación y tendencias entre metodologías ágiles y formales. Metodología utilizada en el Centro de Informatización para la Gestión de Entidades. Revista Serie Científica Vol.4. N 10.
- Jabba, D; Alcocer, A; Rojas, c. 2004. Análisis comparativo de las herramientas de programación Web: PHP, ASP Y JSP, bajo los sistemas operativos Linux y Windows. Ingeniería y desarrollo. 16 ed. Colombia. p 104-115.

- Jiménez, M; Rodríguez, L; Arévalo, O; 2012. Experiencias en la evaluación del proceso aseguramiento de la calidad en el departamento Geoinformática mediante la aplicación de indicadores. Revista Cubana De Ciencias informáticas. Vol. 6. N 3. p 1-14.
- Kasiak, T. y Godoy, D. 2012. Simulación de Proyectos de Software desarrollados con XP: Subsistema de Desarrollo de Tareas. XIV Workshop de Investigadores en Ciencias de la Computación.
- Katz, R. 2009. El papel de las TICS en el desarrollo. 2 ed. España. p 18.
- L'êcuyer J. 2001. La evaluación en la enseñanza superior. Brasil. Paraná. Revista Diálogo Educativo. Vol. 2. p 21-26.
- Marqués, P. 2007. La web 2.0 y sus aplicaciones didácticas. Universidad Autónoma de Barcelona, ES.
- Mindy, S y Erick, T. 2011. Desarrollo de un sistema web basado en ADO.NET para el control de la gestión de inscripción y logística de cursos en la empresa enfoque directo aplicado consultores, C.A. Tesis. Caracas. VEN.
- Moquillaza, S; Huerta, H; Grados, L: 2010. Programación en N capas. Revista de investigaciones de Sistemas e Informática. Universidad Nacional Mayor de San Marcos. Vol 7. p 57-67.
- Moreno, T. 2011. La cultura de la evaluación y la mejora de la escuela. México. Perfiles educativos. Vol. 33. p 116-130.
- Murillo, F y Román M. 2010. Retos en la evaluación de la calidad de la educación en América Latina, revista Iberoamericana de educación. Vol. 53. p. 97-120.
- Naranjo, R; Lopez, S; Guaigua, J. 2013. Análisis comparativo metodoloca en ambientes integrados de desarrollo. Tesis. Ingeniería en sistemas computacionales. Universidad técnica del norte. ECU. p 131.
- Nieves, A. 2009. Metodologías de diseño usada en ingeniería WEB, su vinculación con las NTICS. Tesis. Especialización en TIAE. Universidad Nacional de La Plata. ARG. p 18-30.

- Oliveros, A; Del Valle, S; Wehbe, R. Rousselot, J. 2011. Requerimientos para aplicaciones web. (En línea). ARG. Consultado, 05 de Jun.2013. Disponible en: <http://sedici.unlp.edu.ar/handle/10915/20125>.
- Olmos, M. 2013. Aplicación para móviles de distribución de paquetería. Formato pdf. Disponible en: <http://upcommons.upc.edu/pfc/bitstream/2099.1/18640/1/82453.pdf>
- Pérez, V y Fernández, J. 2005. Las tecnologías de la información y la comunicación en la formación permanente del profesorado de educación física. REIFOP. Disponible en: <http://www.aufop.com>.
- Ponce, T y Fernández, P. 2012. Usuarios inteligentes y la web semántica. Revista Cubana De Ciencias informáticas, Vol. 6. N 1. p 1-10.
- Pressman, R. 2010. Ingeniería de Software un enfoque práctico. México, Mc Graw Interamericana de Editores.
- Quintero, B; Anaya, R; Marín, J. 2012. Un estudio comparativo de herramientas para el modelado con UML. Revista Universidad EAFIT, Vol 41, 60-76.
- Reyna, Y; Pupo, O; Rodríguez, M; Jiménez, A; 2012. Modelos para el diseño de bases de datos difusas. Revista Cubana de Ciencias Informáticas, Vol. 6. p 82-87.
- Rondón, Y; Domínguez, L; Berenguer, A; 2011. Diseño de la base de datos para sistemas de digitalización y gestión de medias. Revista de Informática Educativa y Medios Audiovisuales, Vol. 8, 17-25.
- Rosero, F; López, A; Matamoros, S; 2009. Sistema de control y evaluación de exámenes teóricos y prácticos para los cursos de la academia Microsoft-ESPOL. Guayaquil, Ec. Revista tecnológica ESPOL.
- Rubio, M; Galván, C.; Rodríguez, J; 2013. Propuesta didáctica para el uso de portafolios digitales en educación superior. Revista Electrónica de Tecnología Educativa. Barcelona. ES. Vol 43. p 2-13

- Santos, L; Galan, J; Izquierdo, L; Del Olmo, R. 2009. Aplicaciones de las TIC en el nuevo modelo de enseñanza del EEES. (En línea). ES. Consultado, 05 de Jun. 2013. PDF. Disponible en: revistadyo.com/index.php
- Shapiro, J. 2009. Microsoft SQL Server 2005: The complete reference. McGraw-Hill.
- Suárez, C. 2010. e-Learning as an object of study. Revista de Universidad y Sociedad del Conocimiento. Catalunya. ES. Vol. 1. p 1-11.
- Suárez, E. 2011. Análisis comparativo de los Frameworks EJB3 y ADO. NET Entity framework para el desarrollo de aplicaciones empresariales. Tesis. Escuela Politécnica Nacional, QUITO.
- Valiente, I. y González, H. 2013. Didáctica del proceso de formación de los profesionales asistido por las tecnologías de la información y la comunicación. Revista Pedagogía Universitaria, Vol. 10. p 1-6.
- Vanegas, C. 2013. Funcionalidad del lenguaje integrado de consultas (linq), con ejemplos en visual basic.net. Revista Vínculos, Vol. 8, p 60-69.
- Zapata, C. y Tamayo, P. 2009. Generación del diagrama de casos de uso a partir del lenguaje natural o controlado: Una revisión crítica. Vol. 76. p 193-203.
- Zapata, C; Arango, F. 2012. Los modelos verbales en lenguaje natural y su utilización en la elaboración de esquemas conceptuales para el desarrollo de software: una revisión crítica. Revista Universidad EAFIT, vol. 41, no 137, p. 77-95.

ANEXOS

Anexo 1.

DOCUMENTO DE ENTREGA DE REQUERIMIENTOS.

Documento de requerimientos del sistema

1 Introducción

1.2. Alcance del documento

Este documento describe los requerimientos de software del sistema informático de test on-line, cuyo objetivo principal es brindar una herramienta informática que permita evaluar a los estudiantes de la ESPAM MFL.

Este documento de requerimientos de software es la base del desarrollo de software del proyecto. Describe los siguientes tópicos: [especifique la lista de tópicos]. Este documento no describe [especifique lo que no se describe en este documento].Glosario, (incluir Acrónimos y Abreviaturas)

1.3. Glosario

- RF: Requerimiento funcional.
- RNF: Requerimiento no funcional.

2 Descripciones de Usuarios/Roles

- Administrador.
 - Tendrá los privilegios para crear, modificar, configurar, eliminar y consultar registros de todas las tareas del sistema.
- Estudiante.

- Tendrá los privilegios exclusivamente para rendir una evaluación previamente configurada por el administrador.
- Docentes.
 - Tendrá privilegios para rendir una evaluación previamente configurada por el administrador.

3 Requerimientos Funcionales

Los requerimientos funcionales de un sistema describen la funcionalidad o los servicios que se espera que éste provea. Son entendidos como capacidades que debe exhibir una aplicación con el fin de resolver un problema.

- **Requerimientos de datos o información:** El sistema guardara información correspondiente a reactivos, datos de evaluaciones, e información necesaria por el comité acreditación institucional resultante de dichas evaluaciones.
- **Requerimientos de interfaz (con el usuario):** El sistema debe permitir a un usuario sin mayor experiencia en manejo de sistemas realizar la gestión de una evaluación, desde la creación, configuración, evaluación y reportes. La interfaz debe ser intuitiva que permita a los estudiantes rendir un test sin una explicación prueba, los iconos deben ir relacionados a la tarea que se ejecuta al dar clic sobre ellos.
- **Requerimientos de navegación:** El sistema debe mostrar un menú de navegación fácil e intuitiva.
- **Requerimientos de personalización,** describen cómo debe adaptarse el sistema en función de qué tipo de usuario interactúe con él y de la descripción actual de dicho usuario.

- **Requerimientos transaccionales o funcionales internos**, recogen qué debe hacer el sistema de forma interna, sin incluir aspectos de interfaz o interacción. También son conocidos en el ambiente web como requerimientos de servicios.

ID del Requerimiento:	RF-1			
Nombre del Requerimiento:	Gestión de Áreas.			
Características:	Requerimiento de datos e información.			
Descripción del requerimiento:	Ingresar, editar, eliminar y consultar áreas relacionadas a una asignatura específica.			
Prioridad del requerimiento:	Baja	<u>Alta</u>	Media Alta	Media
				Media Baja

ID del Requerimiento:	RF-2			
Nombre del Requerimiento:	Gestión de Temas.			
Características:	Requerimiento de datos e información.			
Descripción del requerimiento:	Ingresar, editar, eliminar y consultar temas relacionadas a un área específica.			
Prioridad del requerimiento:	<u>Alta</u>	Media Alta	Media	Media Baja
				Baja

ID del Requerimiento:	RF-3			
Nombre del Requerimiento:	Gestión de Preguntas.			
Características:	Requerimiento de datos e información.			
Descripción del requerimiento:	Ingresar, editar, eliminar, consultar y gestionar diferentes tipos de preguntas relacionadas a un tema específico.			
Prioridad del requerimiento:				
<u>Alta</u>	Media Alta	Media	Media Baja	Baja

ID del Requerimiento:	RF-4			
Nombre del Requerimiento:	Gestión de Evaluaciones.			
Características:	Requerimiento de datos e información.			
Descripción del requerimiento:	Ingresar, editar, eliminar, consultar y configurar evaluaciones.			
Prioridad del requerimiento:				
<u>Alta</u>	Media Alta	Media	Media Baja	Baja

ID del Requerimiento:	RF-5			
Nombre del Requerimiento:	Gestión de módulo de evaluación.			
Características:	Requerimiento de datos e información.			
Descripción del requerimiento:	Modulo que permitirá a los estudiantes y docentes rendir una evaluación.			
Prioridad del requerimiento:	<u>Alta</u>	Media Alta	Media	Media Baja
	Baja			

ID del Requerimiento:	RF-6			
Nombre del Requerimiento:	Gestión de Usuarios.			
Características:	Requerimiento de datos e información.			
Descripción del requerimiento:	Permitir el acceso al módulo correspondiente al tipo de usuario que este ingresando.			
Prioridad del requerimiento:	<u>Alta</u>	Media Alta	Media	Media Baja
	Baja			

ID del Requerimiento:	RF-7			
Nombre del Requerimiento:	Gestión de Reportes.			
Características:	Requerimiento de datos e información.			
Descripción del requerimiento:	Generar reportes por carreras, por semestres.			
Prioridad del requerimiento:	Baja	<u>Alta</u>	Media Alta	Media
				Media Baja

En constancia a la aceptación de la entrega de requerimientos presentados firman:

Sr. Adrian Alcivar
Desarrollador.

Ing. Ángel Vélez
Coordinador de evaluación institucional.

Proyecto: Sistema informático de test on-line para la comisión de evaluación de la ESPAM MFL

Fecha: 08/11/2013

Versión: 08/11/2013-01

ANEXO 2.

TEST DE USABILIDAD DE LAS INTERFACES DE USUARIO DEL SISTEMA.

ENCUESTA DE USABILIDAD DEL SISTEMA DE TEST EN LINEA DE LA ESPAM MFL.

Objetivo: Evaluar la interfaz y funcionalidad grafica del sistema web.

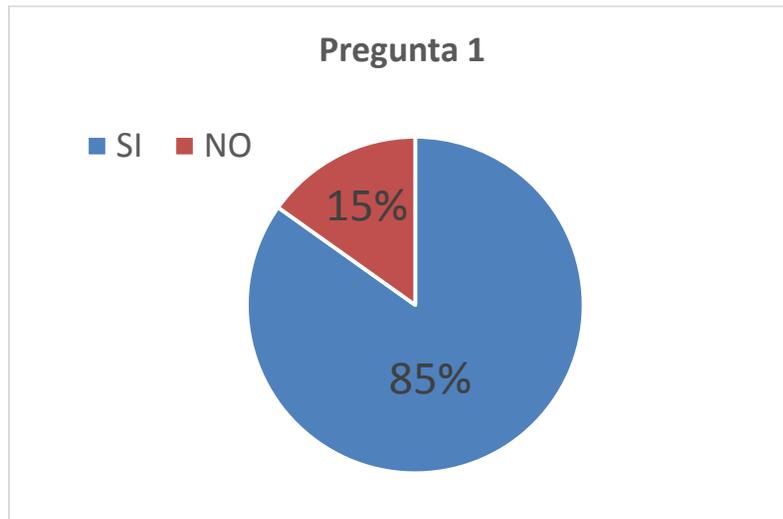
Nº	PREGUNTAS	SI	NO
1	¿El sistema le permite realizar las tareas necesarias?		
2	¿La navegación a través del sistema resulta fácil?		
3	¿La apariencia general del sistema es agradable?		
4	¿Los iconos de la aplicación, identifican la función asignada?		
5	¿La estructura y organización de la información del sistema está bien distribuida?		
6	¿Te han parecido claros y representativos los nombres y descripciones que aparecen en el sistema?		
7	En general. ¿Te fue fácil realizar las tareas que necesitabas?		
8	¿Piensa que vas a necesitar a alguien que le ayude a manejar este programa?		

ANEXO 3.

**ANALISIS DE RESULTADOS DE TEST DE USABILIDAD DE LAS INTERFACES
DE USUARIO DEL SISTEMA.**

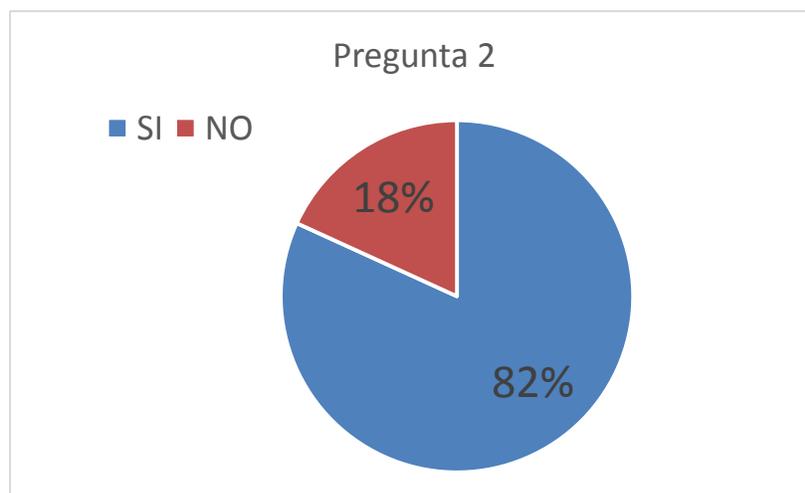
RESULTADOS

PREGUNTA 1. ¿El sistema le permite realizar las tareas necesarias?



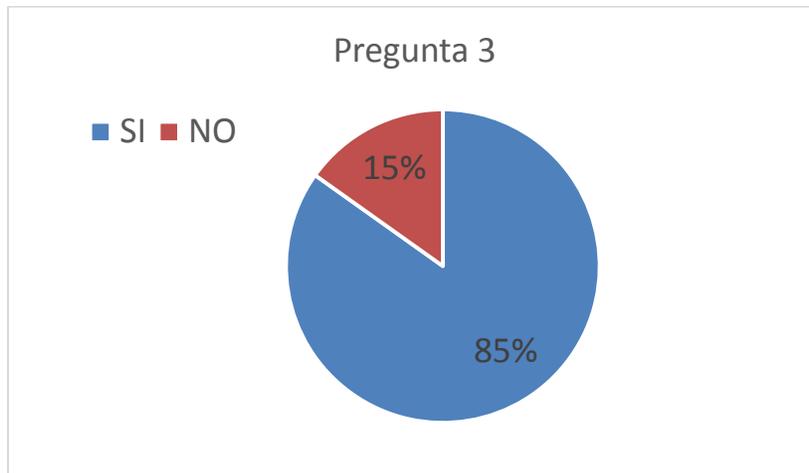
La tabulación de resultados permite concluir que el 85% de los encuestados consideran que el sistema permite realizar las tareas necesarias para rendir la evaluación.

PREGUNTA 2. ¿La navegación a través del sistema resulta facil?.



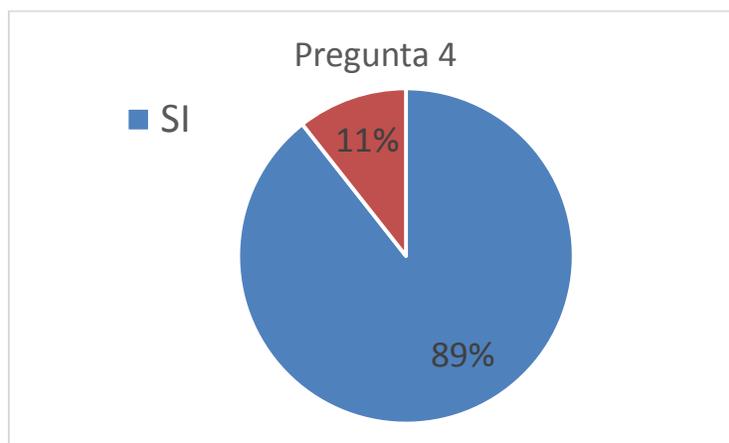
El 82% de los encuestados expresaron que la estructura de navegación del sistema resulta fácil.

PREGUNTA 3. ¿La apariencia general del sistema es agradable?



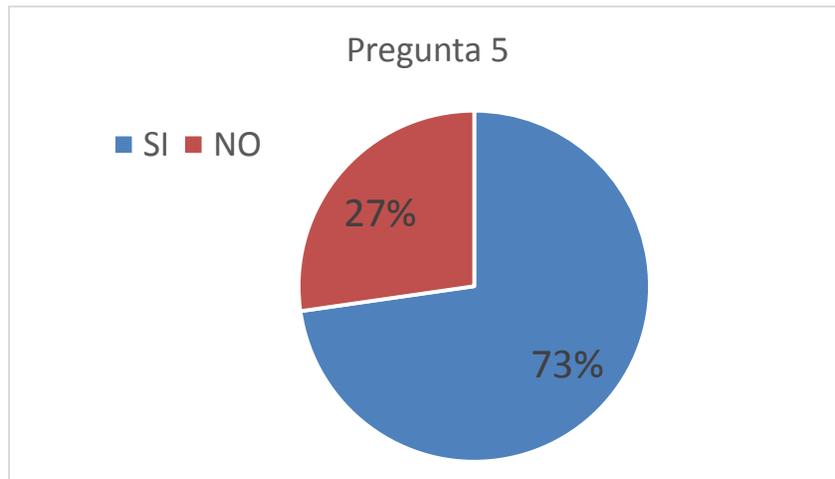
El 85% de la muestra tabulada considera que la apariencia general del sistema es agradable.

PREGUNTA 4. ¿Los iconos de la aplicación, identifican la función asignada?



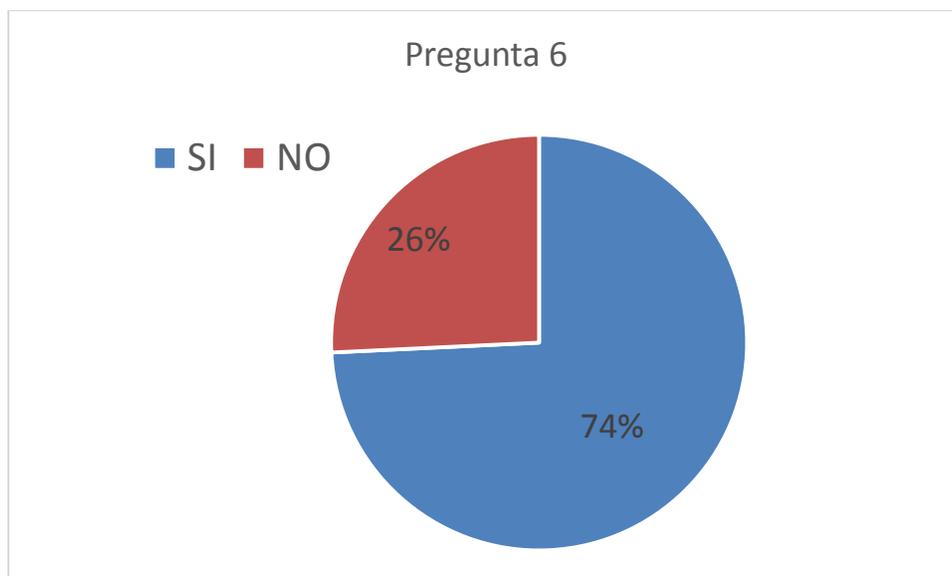
El 89% de la muestra preguntada, manifiesta que los iconos del sistema representan la función o tarea que realiza.

PREGUNTA 5. ¿La estructura y organización de la información del sistema está bien distribuido?



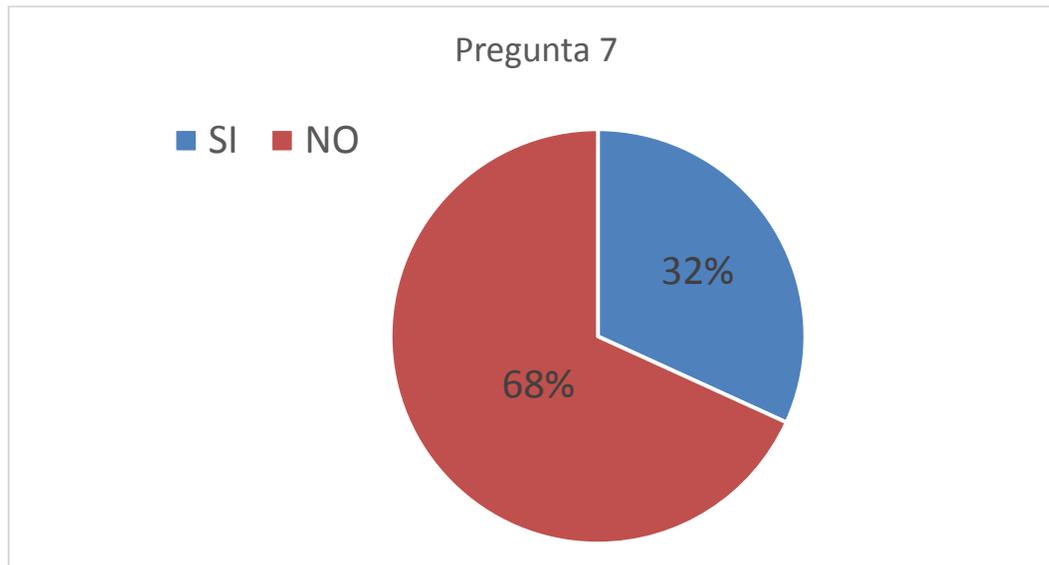
El 73% del sistema manifiesta que la organización de la información en el sistema distribuida de manera uniforme.

PREGUNTA 6. En general, ¿Te fue fácil realizar las tareas que necesitabas?



El 74% de los usuarios del sistema manifestaron que les fue fácil realizar las tareas que necesitaba para concluir una evaluación.

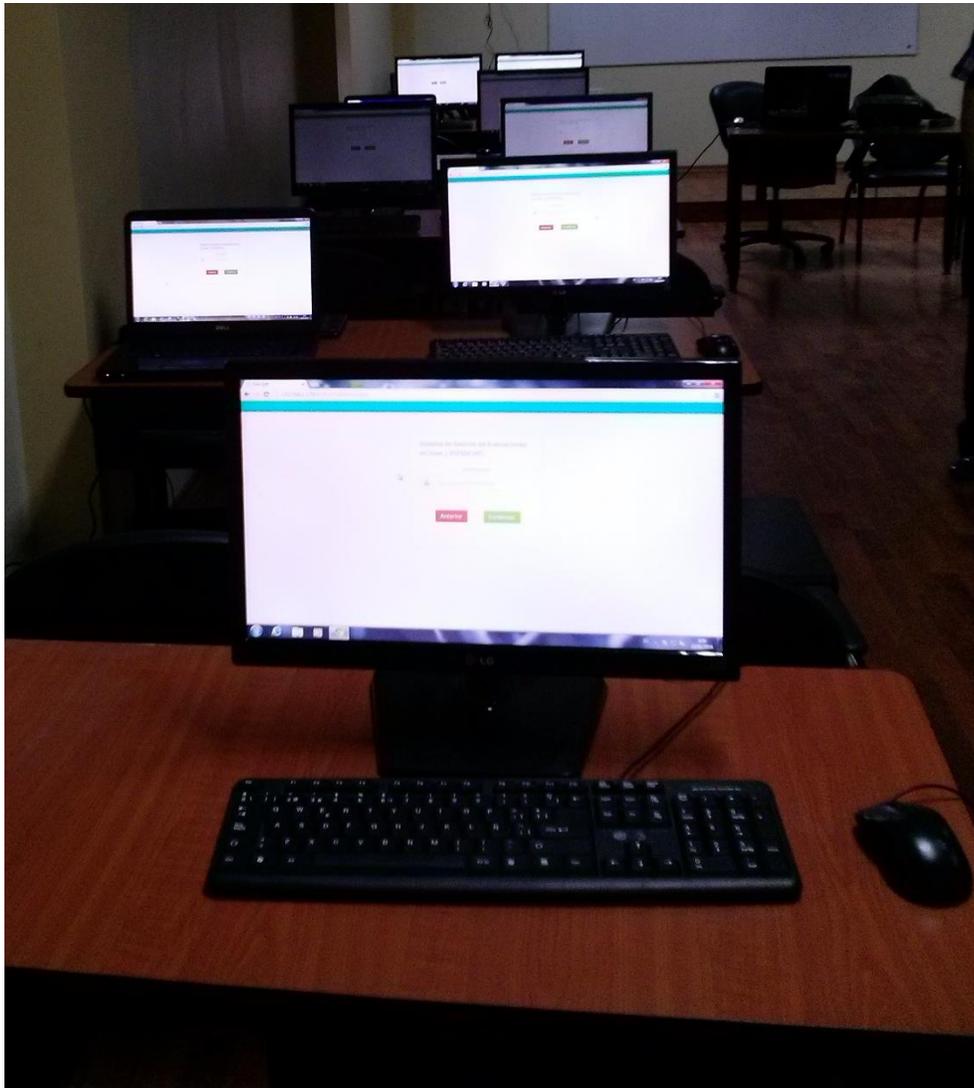
PREGUNTA 7. ¿Piensa que vas a necesitar a alguien que le ayude a manejar este programa?



El 68% manifestó que no necesitara ayuda al momento de rendir una evaluación.

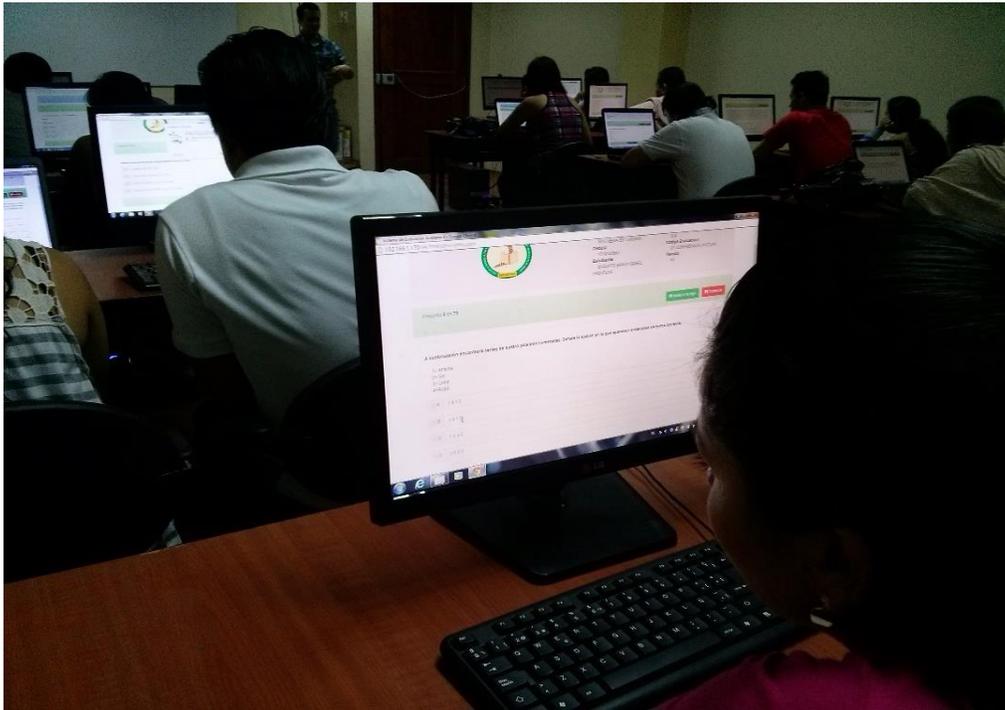
ANEXO 5.
FOTO DEL SISTEMA IMPLEMENTADO EN EL LABORATORIO 201 DE LA
CARRERA DE INFORMATICA

Implementación del sistema web en el laboratorio 201 de la carrera de informática de la ESPAM MFL.



ANEXO 6.
EVALUACION A ESTUDANTES DE LA ESPAM MFL.

EVALUACION A ESTUDIANTES DE LA CARRERA DE TURISMO



EVALUACION A ESTUDIANTES DE LA CARRERA DE AGROINDUSTRIAS



ANEXO 7.
MANUAL DE PROGRAMACIÓN.



MANUAL DE PROGRAMACIÓN

El presente manual busca servir como guía técnica del sistema de evaluaciones en línea de la ESPAM MFL, detalla los métodos y tecnologías de programación utilizados durante el desarrollo con la finalidad de permitir el mantenimiento y sostenimiento del sistema a través del tiempo.

El sistema se lo desarrollo bajo el lenguaje de programación C# y Visual Studio como IDE (Entorno de Desarrollo Integrado). Se trabajó bajo el paradigma de la programación orientada a objetos y se dividió la solución del sistema en tres capas (ver Figura 1).

- Capa de datos.
- Capa de Negocio.
- Capa de Presentación.

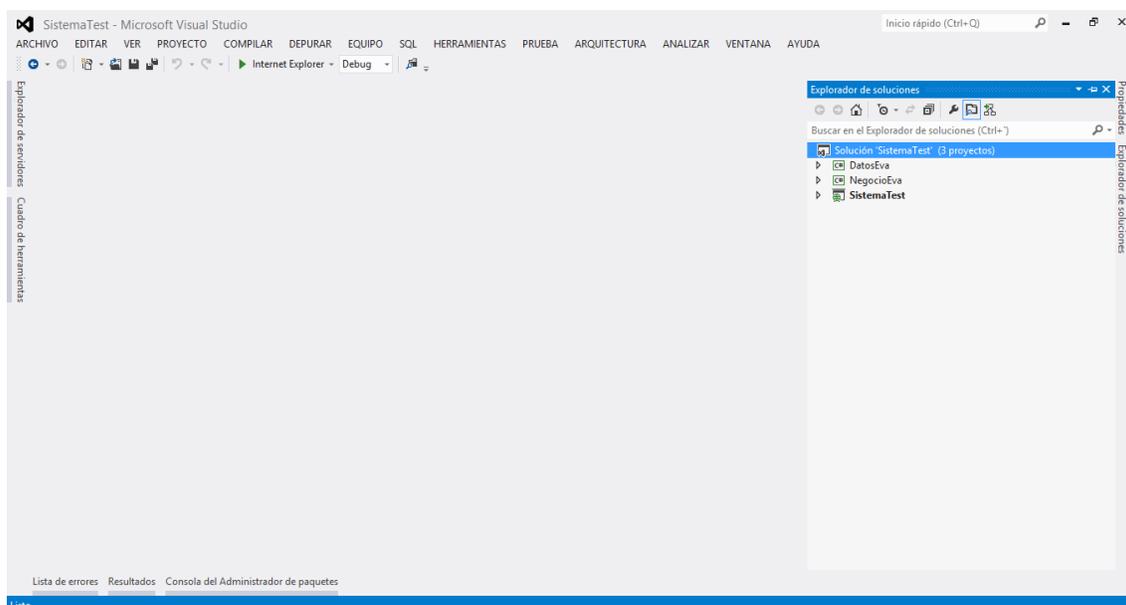


Figura.1.3.1. Captura de pantalla de la solución con las tres capas del proyectos.

CAPA DE ACCESO A DATOS

En la capa de acceso a datos la conexión se la gestionó utilizando Entity Framework V5.0.0 como lo muestra la figura 2, para poder tratar a los elementos de la base de datos como objetos, pues, la base de datos es relacional y el lenguaje de programación es orientado a objetos, esto permitió agregar a la aplicación los procedimientos necesario para la inserción, modificación, eliminación y consultas de los datos.

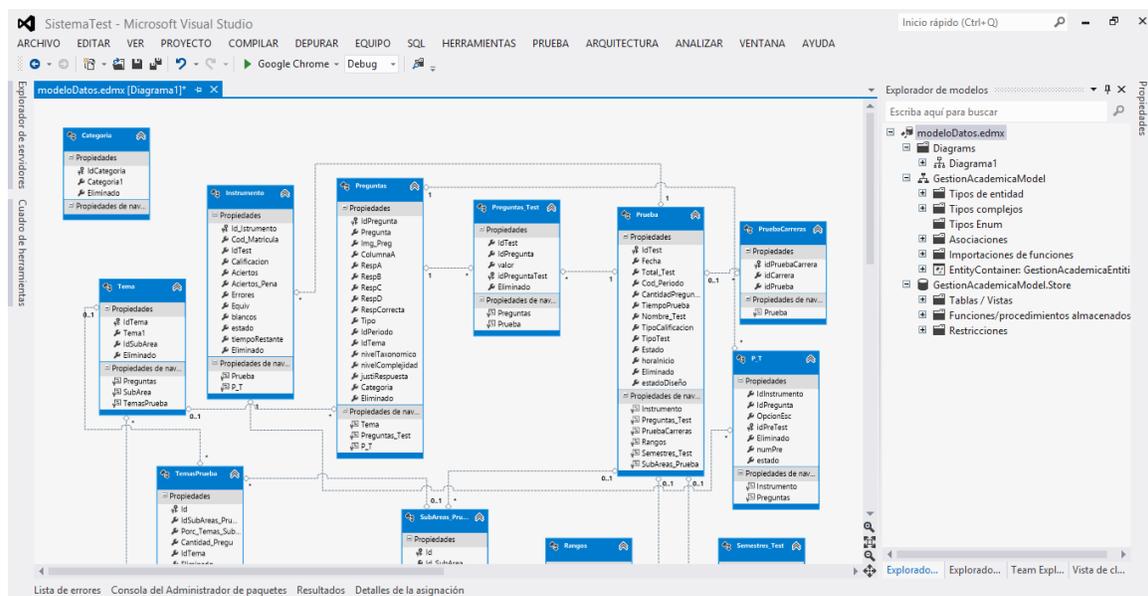


Figura 2. Modelo generado mediante Entity Framework a partir de la base de datos.

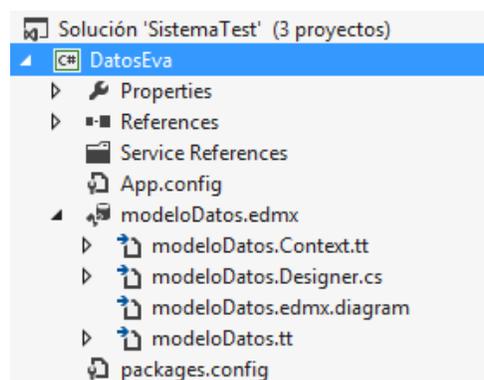


Figura 3. Capa de datos con el modelo generado mediante entity.

CAPA DE NEGOCIOS

En la capa de negocios se hizo uso de librerías de clases, en el lenguaje de programación C# y bajo el paradigma de la programación orientada a objetos, para construir los métodos necesarios para manipular la lógica de la aplicación, se debió considerar agregar las referencias de la capa de datos para hacer posible la utilización de los métodos que realizan la gestión con la base de datos.

Se crearon clases que tendrán la función de clases tontas.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace NegocioEva
{
    public class Prueba
    {
        //Atributos que representan los campos de la tabla Prueba
        public int IdTest { get; set; }
        public Nullable<System.DateTime> Fecha { get; set; }
        public Nullable<int> Total_Test { get; set; }
        public Nullable<int> Cod_Periodo { get; set; }
        public Nullable<int> CantidadPreguntas { get; set; }
        public Nullable<int> TiempoPrueba { get; set; }
        public string Nombre_Test { get; set; }
        public string TipoCalificacion { get; set; }
        public string TipoTest { get; set; }
        public string Estado { get; set; }
        public Nullable<System.TimeSpan> horaInicio { get; set; }
        public string estadoDiseno { get; set; }

        //Constructores en blanco
        public Prueba() { }

        //Constructor que permite setear un objeto de tipo Prueba
        public Prueba(int _idTest, DateTime? _fecha, int? _totalPrueba, int? _idPeriodo,
            int? _cantidadPreguntas, string _nombrePrueba, string _TipoCalificacion, string
            _TipoTest, string _estado, TimeSpan? _horaInicio, string _estadoDiseno)
        {
            this.IdTest = _idTest;
            this.Fecha = _fecha;
        }
    }
}
```

```

        this.Total_Test = _totalPrueba;
        this.Cod_Periodo = _idPeriodo;
        this.CantidadPreguntas = _cantidadPreguntas;
        this.Nombre_Test = _nombrePrueba;
        this.TipoCalificacion = _TipoCalificacion;
        this.TipoTest = _TipoTest;
        this.Estado = _estado;
        this.horaInicio = _horaInicio;
        this.estadoDiseno = _estadoDiseno;
    }
}
}

```

Al trabajar con programación orientada a objetos fue necesario crear clases tontas para todas las tablas de la base de datos. De la misma manera se realizaron los catálogos que tienen los métodos necesarios para la inserción, modificación, eliminación y consulta de registros.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using DatosEva;
//Es necesario incluir la capa datos para tener acceso al Entity generado en la
clase datos.

namespace NegocioEva
{
    public class CatalogoPrueba
    {
        //Objeto de clase GestionAcademicaEntities que permite tener disponible
los procedimientos de la base de datos
        GestionAcademicaEntities Objeto= new GestionAcademicaEntities();

        //Método que permite obtener las pruebas ingresadas en el sistema
        public List<Prueba> ObtenerPrueba()
        {
            List<Prueba> lista = new List<Prueba>();

            foreach (var iten in Objeto.spObtenerTest())
            {
                lista.Add(new Prueba(iten.IdTest, iten.Nombre_Test,
Convert.ToDateTime(iten.Fecha), iten.TiempoPrueba, iten.Total_Test,
iten.CantidadPreguntas, iten.TipoTest, iten.TipoCalificacion, iten.Estado,
iten.estadoDiseno));
            }
            return lista;
        }
    }
}

```

```

//Método que permite insertar un nuevo registro
public int InsertarPrueba(Prueba _dato)
{
    int id;
    try
    {
        id = Convert.ToInt32(Objeto.spInsertarTest(_dato.Fecha,
        _dato.Total_Test, _dato.Cod_Periodo, _dato.Nombre_Test, _dato.TipoCalificacion,
        _dato.TiempoPrueba, _dato.CantidadPreguntas, _dato.TipoTest, _dato.Estado,
        _dato.horaInicio).Select(c => c.Value).First());
    }
    catch (Exception)
    {
        mensaje = "Error al insertar registro";
        throw;
    }
    return id;
}

//Método que permite modificar un registro
public string ModificarPrueba(Prueba _dato)
{
    string mensaje = "Datos modificados exitosamente";
    try
    {
        Objeto.spActualizarTest(_dato.IdTest, _dato.Fecha,
        _dato.Total_Test, _dato.Cod_Periodo, _dato.Nombre_Test, _dato.TipoCalificacion,
        _dato.TiempoPrueba, _dato.CantidadPreguntas, _dato.TipoTest, _dato.Estado,
        _dato.horaInicio);
    }
    catch (Exception)
    {
        mensaje = "Error al modificar registro";
        throw;
    }
    return mensaje;
}

//Metodo que permite eliminar un registro
public string EliminarPrueba(int _datoId)
{
    string mensaje = "Datos eliminados exitosamente";
    try
    {
        Objeto.spEliminarTest(_datoId);
    }
    catch (Exception)
    {
        mensaje = "Error al eliminar registro";
        throw;
    }
}

```

```

        return mensaje;
    }
}
}

```

Los métodos descritos en el apartado anterior permiten la interacción entre las capas de datos y presentación.

CAPA DE PRESENTACIÓN

En la capa de presentación se utilizaron `[WebMethod]` que permiten mantener un método disponible en la web para poder ser ejecutados en el caso del sistema planteado mediante Ajax.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.Services;
using NegocioEva;
//Es necesario incluir la capa negocio para tener acceso a los métodos creados.

namespace SistemaTest
{
    public partial class frmCrudPrueba : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        //Método que permite insertar un nuevo registro, recibe un objeto de tipo Prueba
        //que se llenara con los datos desde JavaScript, y se enviara al método en la
        //clase negocio para su procesamiento.
        [WebMethod]
        public static string insertarPrueba(Prueba parDato)
        {
            //variable que retornara el mensaje de éxito o error de la inserción
            string mensaje = "";
            //Objeto de tipo CatalogoPrueba que nos permiten tener disponibles los métodos
            //de inserción, modificación, eliminación y consulta creados en el Catalogo.
            CatalogoPrueba _objeto = new CatalogoPrueba();
            try
            {
                mensaje = _objeto.InsertarPrueba(parDato);
            }
        }
    }
}

```

```

        catch (Exception)
        {
            mensaje = "Error en la inserción de los datos";
            throw;
        }
        return mensaje;
    }

//Método que permite actualizar un registro
[WebMethod]
public static string actualizarPrueba(Prueba parDato)
{
    string mensaje = "";
    CatalogoPrueba _objeto = new CatalogoPrueba();
    try
    {
        mensaje = _objeto.ModificarPrueba(parDato);
    }
    catch (Exception)
    {
        mensaje = "Error en la modificación de los datos";
        throw;
    }
    return mensaje;
}

//Método que permite eliminar un registro
[WebMethod]
public static string eliminarPrueba(int parIdPrueba)
{
    string mensaje = "";
    CatalogoPrueba _objeto = new CatalogoPrueba();
    try
    {
        mensaje = _objeto.EliminarPrueba(parIdPrueba);
    }
    catch (Exception)
    {
        mensaje = "Error en la eliminación de los datos";
        throw;
    }

    return mensaje;
}

//Método que permite obtener una lista de los registros existentes
[WebMethod]
public static List<Prueba> ObtenerPrueba()
{
    CatalogoPrueba _objeto = new CatalogoPrueba();
    List<Prueba> lista = new List<Prueba>();

    lista = _objeto.ObtenerPruebaString();
}

```

```

        return lista;
    }

//Método que permite obtener un registro por id
[WebMethod]
public static List<Prueba> ObtenerPruebaPorId(int parIdPrueba)
{
    CatalogoPrueba _objeto = new CatalogoPrueba();
    List<Prueba> lista = new List<Prueba>();

    lista = _objeto.ObtenerPruebaString().Where(c => c.IdTest ==
parIdPrueba).ToList();

    return lista;
}

[WebMethod]
}
}
}

```

Los métodos creados anteriormente son ejecutados mediante JavaScript y Jquery disponible en: <http://jquery.com> como se detalla a continuación.

```

function GuardarRegistro() {

//Método que valida mediante JavaScript que no se ingresen datos en blanco
    if($("#ddlPeriodo").val() == "" || $("#ddlPeriodo").val() == null) {
        alert("Porfavor, todos los datos son necesarios");
        $("#ddlPeriodo").focus();
        return false;
    }

//Método que setea el parámetro recibido por el WebMethod
    var data = { parDato: {
        Fecha: $("#txtFecha").val(),
        Total_Test: $("#txtCalificacion").val(),
        Cod_Periodo: $("#ddlPeriodo").val(),
        CantidadPreguntas: $("#txtNumPre").val(),
        TiempoPrueba: $("#txtTiempo").val(),
        Nombre_Test: $("#txtNombre").val(),
        TipoCalificacion: $("#ddlTipoCalificacion").val(),
        TipoTest: $("#ddlTipoPrueba").val(),
        Estado: $("#ddlEstado").val(),
        horaInicio: $("#TextHoraInicio").val()
    }
};
}

```

```

//Funcion Ajax que permite enviar y ejecutar el método
$.ajax({
  url: "frmCrudPrueba.aspx/insertarPrueba", //Dirección del servicio web
  data: JSON.stringify(data), //Envía los parámetros al método web
  dataType: "json", //Parámetro que permite que los datos lleguen en json
  method: "POST",
  contentType: "application/json; charset=utf-8",
  success: //Recibe la respuesta del método llamado
    function (result) {
      paginacion(1);
      $('#modalfrm').modal('hide');
    }
});
return false;
});
}

```

El siguiente código nos permite llenar la tabla con los registros existentes en el sistema.

```

var llenarTabla = function () {
  limpiarTabla();
  $.ajax({
    type: "POST",
    url: "frmCrudPrueba.aspx/llenarTablaLista", //Direccion del servicio
web seguido de /Nombre del metodo a llamar
    data: "{}", //Esto se utiliza si deseamos pasar algun parametro al
metodo del servicio web ejm: {'identificacion':'1234'}
    contentType: "application/json; charset=utf-8",
    dataType: "json", //Esto quiere decir que los datos nos llegaran
como un objeto json
    success: function (response) { //Aca se recibe la respuesta del
metodo llamado

      //llenamos el objeto datos_clientes con la lista de clientes que
nos retorno el metodo del servicio web
      var datos_clientes = (typeof response.d) == 'string' ? eval('(' +
response.d + ')') : response.d;

      //Recorremos los registros
for (var i = 0; i < datos_clientes.length; i++) {
  var IdTest = datos_clientes[i].IdTest;
  var Nombre = datos_clientes[i].Nombre_Test;
  var Fecha = datos_clientes[i].fechaString;
  var Tiempo = datos_clientes[i].tiempoPrueba_string;
  var Valor = datos_clientes[i].Total_Test;
  var Preguntas = datos_clientes[i].CantidadPreguntas;
  var Tipo = datos_clientes[i].TipoTest;
  var Calificacion = datos_clientes[i].TipoCalificacion;
  var Estado = datos_clientes[i].Estado;

  //Escribimos los datos mediante jquery en la tabla #registros que creamos
previamente en la pagina web

```

```

$("#registros").append("<tr><td>" + Nombre + "</td>" +
    "<td>" + Fecha + "</td>" +
    "<td>" + Tiempo + "</td>" +
    "<td>" + Valor + "</td>" +
    "<td>" + Preguntas + "</td>" +
    "<td>" + Tipo + "</td>" +
    "<td>" + Calificacion + "</td>" +
    "<td>" + Estado + "</td>" +
    "<td><a class='btn btn-small btn-success' onclick='poblarFormulario(" +
    IdTest + ")'><i class='btn-icon-only icon-ok'></i>Ver</a>&nbsp;<a
    class='btn btn-danger btn-small' onclick='eliminarRegistro(" + IdTest +
    ")'><i class='btn-icon-only icon-remove'></i>Quitar</a>&nbsp;<a
    class='btn btn-small btn-info' onclick='activarTest(" + IdTest + ")'><i
    class='btn-icon-only icon-ok'></i>Activar</a>&nbsp;<a class='btn btn-
    small btn-warning' onclick='agregarRangos(" + IdTest + ")'><i class='btn-
    icon-only icon-asterisk'></i>Agregar Rangos</a>&nbsp;<a class='btn btn-
    small btn-inverse' onclick='agregarSemestres(" + IdTest + ")'><i
    class='btn-icon-only icon-list-alt'></i>Agregar Semestres</a></td></tr>"
);
    }
});
};

```

Es necesario mencionar la necesidad de incluir las librerías necesarias para la ejecución e interacción del usuario.

```

<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-
scale=1.0, user-scalable=no" />
<meta name="apple-mobile-web-app-capable" content="yes" />
<link href="css/bootstrap.min.css" rel="stylesheet" />
<link href="css/bootstrap-responsive.min.css" rel="stylesheet" />
<link href="css/font-awesome.css" rel="stylesheet" />
<link href="css/style.css" rel="stylesheet" />
<link href="css/pages/dashboard.css" rel="stylesheet" />
<script src="js/jquery-1.9.0.js"></script>
<script src="js/bootstrap.min.js"></script>

```

El entorno grafico del sistema web se la realizó utilizando bootstrap como framework con herramientas disponibles para la maquetación web disponible en <http://getbootstrap.com/>.

El siguiente código muestra la estructura base del framework.

```

<div class="main">
    <div class="main-inner">
        <div class="container">
            <div class="row">

```

```
<div class="span12">
    <!--Área de contenido -->
</div>
<!-- /span12 -->
</div>
<!-- /row -->
</div>
<!-- /container -->
</div>
<!-- /main-inner -->
</div>
<!-- /main -->
```

ANEXO 7.
MANUAL DE USUARIO.

**SISTEMA INFORMÁTICO DE TEST
ON-LINE | ESPAM MFL**
Manual de instrucciones

SISTEMA INFORMÁTICO DE TEST ON-LINE | ESPAM MFL

<< Manual de usuario >>

El Sistema de evaluaciones en línea, es un software desarrollado con la finalidad de facilitar y optimizar el proceso de evaluación de los y las estudiantes de la Escuela Superior Politécnica Agropecuaria de Manabí. El presente documento servirá como guía para la utilización del sistema generador de test ON-LINE y detalla los pasos a seguir para el ingreso, modificación, eliminación, consulta y configuración de los datos necesarios para el correcto funcionamiento del Sistema.

Validación de Usuario

Ingreso al sistema:

1. Seleccionar el tipo de usuarios a validarse en el sistema según los privilegios asignados por el administrador.
2. Ingresar usuario.
3. Ingresar Contraseña.
4. Dar clic en el botón "Validar", Se las credenciales ingresadas son correctas, según el tipo de usuario el sistema los direccionará al menú de selección de evaluación o al panel de administración del sistema.

1

Tipo de usuario

Lista los tipos de usuarios permitidos en el sistema web.

2

Caja de texto que permitirá al usuario ingresar el usuario asignado por parte del administrador del sistema.

3

Validar

Si las credenciales correspondientes a usuario y contraseña según el tipo de usuario están correctas, al presionar el botón se re direccionara al área de trabajo correspondiente.

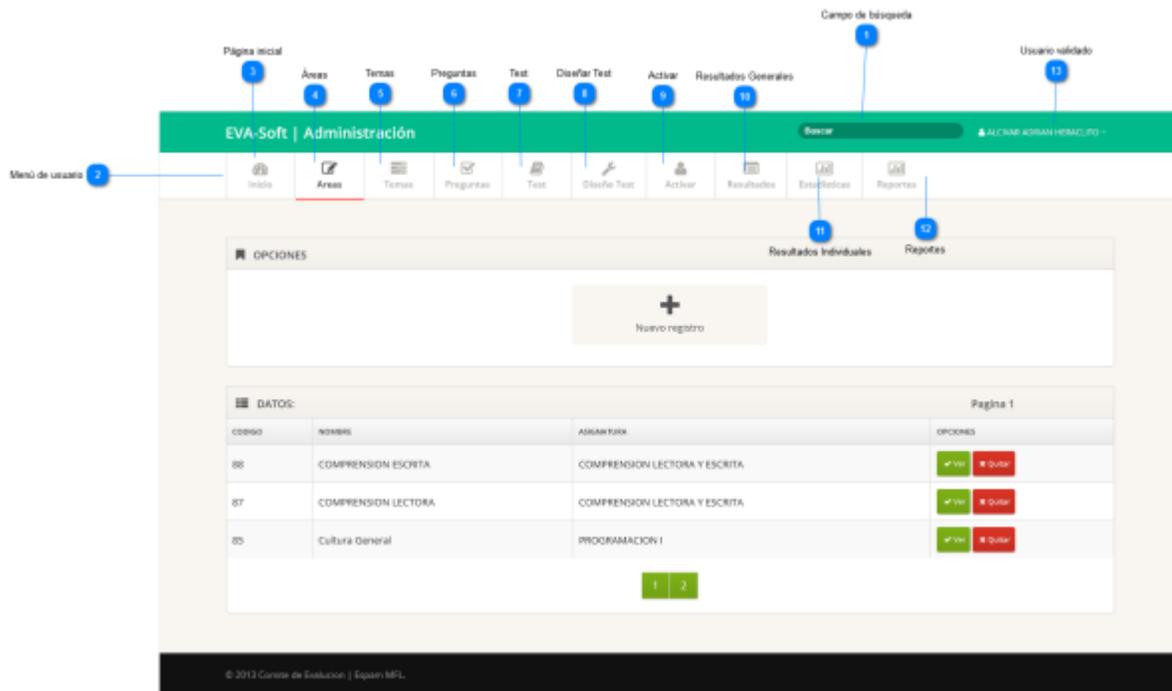


4 Contraseña

Caja de texto que permitir al usuario ingresar la contraseña correspondiente al usuario asignado.

Panel de administración

El presente panel de administración permite al usuario la navegación entre los formularios, muestra un menú con las opciones necesarias y en la parte superior derecha muestra y permite cerrar la sesión al usuario validado actualmente en el sistema.



1

Campo de búsqueda

Elemento que permite realizar búsquedas en el sistema

2

Menú de usuario



Menú de navegación del sistema, contiene los enlaces a las operaciones principales del sistema, para dirigirse a alguna con dar clic es suficiente.

3

Página inicial



Inicio

Muestra la pantalla inicial del sistema, con un mensaje de bienvenida

4

Áreas



Areas

Opción que direccionara al usuario a la pantalla de gestión de áreas del sistema

5

Temas



Temas

Opción que direccionará al usuario a la pantalla de gestión de temas del sistema

6

Preguntas



Preguntas

Opción que direccionará al usuario a la pantalla de gestión de preguntas del sistema

7

Test



Test

Opción que direccionara al usuario a la pantalla de gestión de pruebas del sistema

8

Diseñar Test



Diseño Test

Permite realizar todas las configuraciones

9

Activar

Activar

Redirección al formulario que permite activar los estudiantes validados en una evaluación.

10

Resultados Generales

Resultados

Dirección al usuario a una pantalla que le permite mostrar resultados de una evaluación directamente en el navegador.

11

Resultados Individuales

Estadísticas

Dirección al formulario que muestra resultados de una evaluación filtrados por carreras.

12

Reportes

Reportes

Botón que direccionará a un formulario que permitirá obtener reportes para impresión o exportación.

13

Usuario validado

Panel que muestra el usuario validado actualmente en el sistema. Brinda la opción de cerrar sesión si el usuario así lo deseara.

Gestión de Áreas

Ingreso de nuevo registro:

1. Abrir el formulario correspondiente a las Áreas.
2. Dar clic en el botón "Nuevo registro" (1).
3. Se abrirá la siguiente ventana [Clic aquí](#).
4. Llenar los datos correspondientes.
5. Dar clic en el botón Guardar.

Modificación de un registro:

1. Abrir el formulario correspondiente a las Áreas.
2. Identificar el registro a modificar y dar clic en el botón "Ver" (2).
3. Se abrirá la siguiente ventana [Click aquí](#), con los datos a modificar.
4. Actualizar los datos correspondientes.
5. Dar clic en el botón Actualizar.

Eliminación de un registro:

1. Abrir el formulario correspondiente a las Áreas.
2. Identificar el registro a eliminar y dar clic en el botón "Quitar" (3).
3. Se eliminará el registro seleccionado y se actualizará la tabla.

Consulta de un registro:

1. Abrir el formulario correspondiente a las Áreas.
2. Escribir el criterio de búsqueda en la caja de texto (6).
3. Presionar la tecla Enter.
4. Si existen registros relacionados al criterio ingresado se presentaran en la tabla de los registros con las opciones para modificarlo o eliminarlo de ser el caso.

EVA-Soft | Administración

Inicio Áreas Temas Preguntas Test Diseño Test Activar Resultados Estadísticas Reportes

OPCIONES

Tabla de registros ingresados

Nuevo registro

DATOS: Pagina 1

CODIGO	NOMBRE	ASIGNATURA	OPCIONES
88	COMPRESION ESCRITA	COMPRESION LECTORA Y ESCRITA	Ver Quitar
87	COMPRESION LECTORA	COMPRESION LECTORA Y ESCRITA	Ver Quitar
85	Cultura General	PROGRAMACION I	Ver Quitar

Paginación 1 2

© 2013 Comité de Evaluación | Examen MFL

Pantalla que permite realizar la gestión de las áreas del sistema.

1

Nuevo registro



Muestra una ventana emergente que permite realizar el ingreso de áreas al sistema.

2



Permite modificar el registro al que haga referencia, abrirá la ventana emergente con la opción de actualizar el registro.

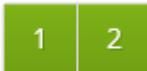
3



Permite eliminar el registro al que haga referencia, actualizará la tabla sin el registro eliminado.

4

Paginación



Muestra una paginación con la finalidad de facilitar la navegación por los registros.

5

Tabla de registros ingresados

DATOS:			Página 1
CODIGO	NOMBRE	ASIGNATURA	OPCIONES
88	COMPRESION ESCRITA	COMPRESION LECTORA Y ESCRITA	<input type="checkbox"/> Ver <input type="checkbox"/> Quitar
87	COMPRESION LECTORA	COMPRESION LECTORA Y ESCRITA	<input type="checkbox"/> Ver <input type="checkbox"/> Quitar
85	Cultura General	PROGRAMACION I	<input type="checkbox"/> Ver <input type="checkbox"/> Quitar

1 2

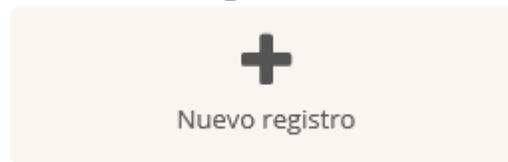
Muestra los registros existentes en el sistema.

6



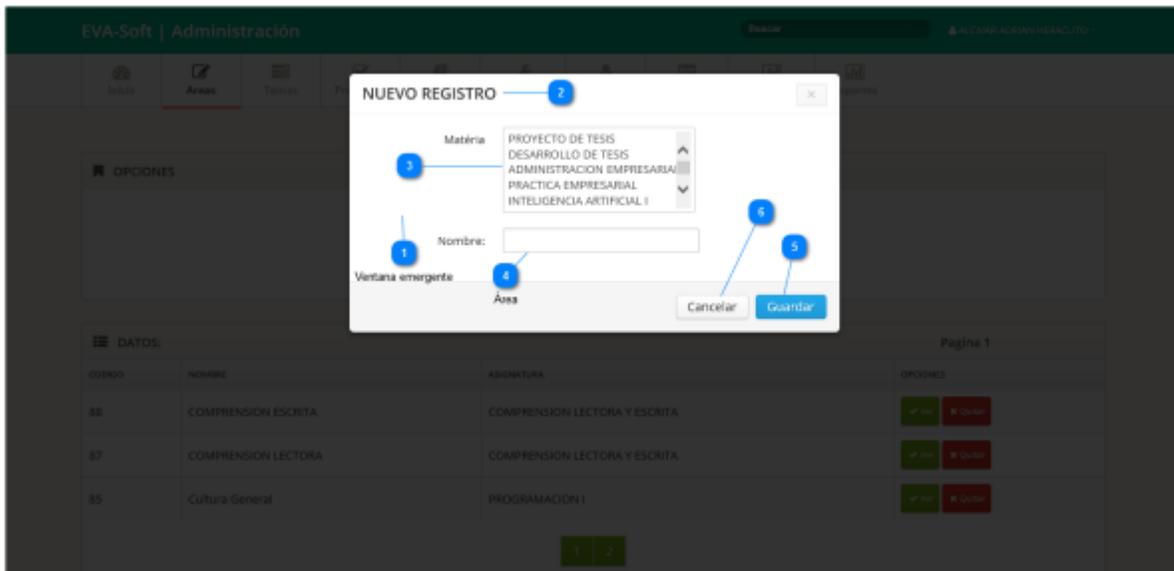
<TAREAS PENDIENTES>: Descripción del control

Nuevo registro



Al dar clic en el botón siguiente ventana:

se mostrara la



1

Ventana emergente

Se mostrará al dar clic en el botón correspondiente a "NUEVO REGISTRO", permite realizar el ingreso y modificación de un nuevo registro.

2 NUEVO REGISTRO

Cabecera que muestra la función de la ventana.

Muestra las asignaturas ingresadas al sistema

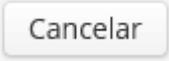
4 Área

Caja de texto que permite ingresar el área a la asignatura previamente seleccionada.

5 Botón guardar

Botón que guardará el nuevo registro a la base de datos, posterior a eso cerrará la ventana emergente.

6

Botón cancelarCancelar

Cierra la ventana emergente.

Gestión de Temas.

Ingreso de nuevo registro:

1. Abrir el formulario correspondiente a Temas.
2. Dar clic en el botón "Nuevo registro" (1).
3. Se abrirá la siguiente ventana [Clic aquí](#).
4. Llenar los datos correspondientes.
5. Dar clic en el botón Guardar.

Modificación de un registro:

1. Abrir el formulario correspondiente a Temas.
2. Identificar el registro a modificar y dar clic en el botón "Ver" (2).
3. Se abrirá la siguiente ventana [Click aquí](#), con los datos a modificar.
4. Actualizar los datos correspondientes.
5. Dar clic en el botón Actualizar.

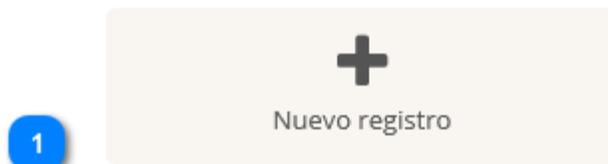
Eliminación de un registro:

1. Abrir el formulario correspondiente a Temas.
2. Identificar el registro a eliminar y dar clic en el botón "Quitar" (3).
3. Se eliminará el registro seleccionado y se actualizará la tabla.

Consulta de un registro:

1. Abrir el formulario correspondiente a Temas.
2. Escribir el criterio de búsqueda en la caja de texto.
3. Presionar la tecla Enter.
4. Si existen registros relacionados al criterio ingresado se presentaran en la tabla de los registros con las opciones para modificarlo o eliminarlo de ser el caso.

Pantalla que permite realizar la gestión de los temas del sistema.



Abre una ventana emergente que permite el ingreso de un nuevo registro.

2 Tabla de registros

DATOS:				Página 1
CODIGO	TEMA	SUBAREA	OPCIONES	
34	temas uno	Sub area dos	<input type="checkbox"/> Ver	<input type="checkbox"/> Quitar
33	ORDENAR FRASES	COMPRESION ESCRITA	<input type="checkbox"/> Ver	<input type="checkbox"/> Quitar
32	SEMÁNTICA	COMPRESION ESCRITA	<input type="checkbox"/> Ver	<input type="checkbox"/> Quitar

Paginación 1 2 3 4 5

Muestra los registros existentes en el sistema, y brinda las opciones para modificarlos y eliminarlos.



Abre una ventana emergente que permite actualizar el registro.



Elimina el registro seleccionado.



Paginación



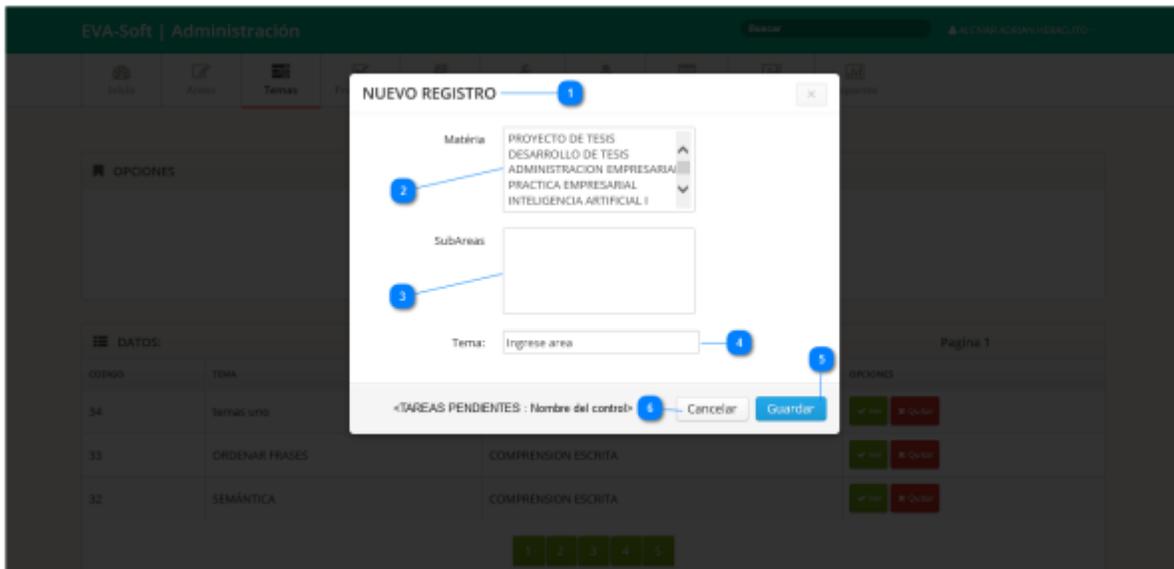
Muestra una paginación con la finalidad de facilitar la navegación por los registros.

Nuevo registro



Al dar clic en el botón siguiente ventana:

se mostrara la



Ventana emergente que se muestra al dar clic en Nuevo Registro

1 NUEVO REGISTRO

Matéria

2

- PROYECTO DE TESIS
- DESARROLLO DE TESIS
- ADMINISTRACION EMPRESARIAL
- PRACTICA EMPRESARIAL
- INTELIGENCIA ARTIFICIAL I

Lista las asignaturas ingresadas en el sistema.

SubAreas

3

Lista las áreas ingresadas a la asignatura seleccionada.

4 Tema:

Caja de texto que permite ingresar la información a guardar al sistema.

5

Guarda el registro a ingresar al sistema.

6

Cierra la ventana emergente.

Gestión de Preguntas

Ingreso de nuevo registro:

1. Abrir el formulario correspondiente a Preguntas.
2. Dar clic en el botón "Nuevo registro" (1).
3. Se abrirá la siguiente ventana [Clic aquí](#).
4. Llenar los datos correspondientes.
5. Dar clic en el botón Guardar.

Modificación de un registro:

1. Abrir el formulario correspondiente a Preguntas.
2. Identificar el registro a modificar y dar clic en el botón "Ver" (2).
3. Se abrirá la siguiente ventana [Click aquí](#), con los datos a modificar.
4. Actualizar los datos correspondientes.
5. Dar clic en el botón Actualizar.

Eliminación de un registro:

1. Abrir el formulario correspondiente a Temas.
2. Identificar el registro a eliminar y dar clic en el botón "Quitar" (3).
3. Se eliminará el registro seleccionado y se actualizará la tabla.

Consulta de un registro:

1. Abrir el formulario correspondiente a Temas.
2. Escribir el criterio de búsqueda en la caja de texto.
3. Presionar la tecla Enter.
4. Si existen registros relacionados al criterio ingresado se presentaran en la tabla de los registros con las opciones para modificarlo o eliminarlo de ser el caso.

EVA-Soft | Administración Buscar ▲ ALCIVAR ADRIAN HERACLITO

Inicio Areas Temas **Preguntas** Test Diseño Test Activar Resultados Estadísticas Reportes

OPCIONES

1 + Nuevo registro 3

DATOS: 2 Tabla de registros existentes en el sistema Página 1

TIPO	TEMA	PREGUNTA	RESPUESTA	NIVEL TAXONÓMICO	NIVEL COMPLEJIDAD	CATEGORIA	OPCIONES
TEXTO	33	A continuación encontrará series de cuatro palabras numeradas. Señale la opción en la que aparecen ordenadas de forma correcta.	C	CONOCIMIENTO	ALTA	ORDENAMIENTO	<input type="checkbox"/> Ver <input type="checkbox"/> Quitar
TEXTO	33	A continuación encontrará series de cuatro palabras numeradas. Señale la opción en la que aparecen ordenadas de forma correcta.	B	CONOCIMIENTO	ALTA	ORDENAMIENTO	<input type="checkbox"/> Ver <input type="checkbox"/> Quitar
TEXTO	33	A continuación encontrará series de cuatro palabras numeradas. Señale la opción en la que aparecen ordenadas de forma correcta.	D	CONOCIMIENTO	ALTA	ORDENAMIENTO	<input type="checkbox"/> Ver <input type="checkbox"/> Quitar

Paginación 5 1 2 3 4 5 6 7 8 9 10 4

© 2013 Comité de Evaluación | Españ MFL



Mostrará un panel con los controles necesarios para hacer el ingreso de las preguntas al sistema

2 **Tabla de registros existentes en el sistema**

DATOS:				Página 1			
TIPO	TEMA	PREGUNTA	RESPUESTA	NIVEL TAXONÓMICO	NIVEL COMPLEJIDAD	CATEGORÍA	OPCIONES
TEXTO	33	A continuación encontrará series de cuatro palabras numeradas. Señale la opción en la que aparecen ordenadas de forma correcta.	C	CONOCIMIENTO	ALTA	ORDENAMIENTO	<input type="checkbox"/> Ver <input type="checkbox"/> Quitar
TEXTO	33	A continuación encontrará series de cuatro palabras numeradas. Señale la opción en la que aparecen ordenadas de forma correcta.	B	CONOCIMIENTO	ALTA	ORDENAMIENTO	<input type="checkbox"/> Ver <input type="checkbox"/> Quitar
TEXTO	33	A continuación encontrará series de cuatro palabras numeradas. Señale la opción en la que aparecen ordenadas de forma correcta.	D	CONOCIMIENTO	ALTA	ORDENAMIENTO	<input type="checkbox"/> Ver <input type="checkbox"/> Quitar

Tabla que muestra las preguntas existentes en el sistema, con las opciones de modificarlas o eliminarlas de ser el caso.



Permite modificar el registro seleccionado.



Elimina la pregunta seleccionada.

5

Paginación



<TAREAS PENDIENTES>: Descripción del control

Nueva pregunta



Al dar clic en el botón siguiente panel:

se mostrara el

El panel de configuración de una nueva pregunta se divide en dos secciones: **GENERALES** y **PREGUNTAS**.

GENERALES:

- 1. Período: Lista desplegable con opciones: SEPTIEMBRE 2013-MARZO 2014, ABRIL - SEPTIEMBRE 2013, OCTUBRE 2012-MARZO 2013, MAYO-OCTUBRE 2012, NOVIEMBRE-ABRIL 2012.
- 2. Materia: Lista desplegable con opciones: INGLÉS II, MECÁNICA DE FLUIDOS I, MECÁNICA DE FLUIDOS II, MATEMÁTICAS II, COMPRENSIÓN LECTORA Y ESCRITA.
- 3. Área: Lista desplegable con opciones: COMPRENSIÓN LECTORA, COMPRENSIÓN ESCRITA.
- 4. Temas: Lista desplegable con opciones: CAUSA Y CONSECUENCIA, INFERENCIA, TEXTOS NARRATIVOS, TEXTOS DESCRIPTIVOS, TEXTO INFORMATIVO.
- 5. Tipo: Lista desplegable con opción: TEXTO.
- 6. Categoría: Lista desplegable con opción: ORDENAMIENTO.

PREGUNTAS:

- 7. Cabecera Pregunta: Campo de texto con el placeholder "Ingrese la cabecera de la pregunta aquí".
- 8. Pregunta: Campo de texto con el placeholder "Ingrese la pregunta aquí".
- 9. Items: Campo de texto con un botón "Agregar" a su derecha.
- AGREGADOS: Tabla con columnas "ID", "ITEM" y "OPCIONES".
- 10. Opción A: Campo de texto con el placeholder "Ingrese la opción A".
- 11. Opción B: Campo de texto con el placeholder "Ingrese la opción B".
- 12. Opción C: Campo de texto con el placeholder "Ingrese la opción C".
- 13. Opción D: Campo de texto con el placeholder "Ingrese la opción D".
- 14. Respuesta correcta: Lista desplegable con opción: A.
- 15. Nivel Taxonomico: Lista desplegable con opción: CONOCIMIENTO.
- 16. Nivel Complejidad: Lista desplegable con opción: ALTA.
- 17. Justificación Respuesta: Campo de texto con el placeholder "Ingrese la justificación de la opción".

<TAREAS PENDIENTES>: Añada aquí la descripción. No olvide añadir una palabra clave para este tema.

1

SEPTIEMBRE 2013-MARZO 2014
 ABRIL - SEPTIEMBRE 2013
 OCTUBRE 2012-MARZO 2013
 MAYO-OCTUBRE 2012
 NOVIEMBRE-ABRIL 2012

Lista los periodos existentes en el sistema.

2

INGLES III
MECANICA DE FLUIDOS I
MECANICA DE FLUIDOS II
MATEMATICAS III
COMPRESION LECTORA Y ESC

Lista las materias existentes para el periodo escogido.

3

COMPRESION LECTORA
COMPRESION ESCRITA

Muestra las áreas ingresadas previamente de acuerdo a la materia escogida.

4

CAUSA Y CONSECUENCIA
INFERENCIA
TEXTOS NARRATIVOS
TEXTOS DESCRIPTIVOS
TEXTO INFORMATIVO

Lista los temas ingresados previamente

5

TEXTO

Muestra las opciones del tipo de pregunta a ingresar, si es de tipo texto o imagen.

6

ORDENAMIENTO

Enlista las categorías de preguntas soportadas por el sistema.

7

Ingrese la cabecera de la pregunta
aqui

Caja de texto disponible para el ingreso de parte de la pregunta correspondiente a la cabecera de la pregunta.

8

Ingrese la pregunta aqui

Caja de texto disponible para el ingreso de parte de la pregunta correspondiente a la interrogación puntual.

9

Caja de texto disponible para el ingreso de ítems dependiendo del tipo de preguntas

10

Ingrese la opcion A

Ingreso de la opción A

11

Ingrese la opcion B

Ingreso de la opción B

12

Ingrese la opcion C

Ingreso de la opción C

13

Ingreso de la opción D

14

Lista de selección donde el usuario debe seleccionar cuales de las opciones ingresadas es la opción correcta.

15

Selección del nivel taxonómico de la pregunta a ingresar.

16

Nivel de complejidad de la pregunta a ingresar.

17

Caja de texto disponible para el ingreso de la justificación de la pregunta a ingresar.

Gestión de Test

Ingreso de nuevo registro:

1. Abrir el formulario correspondiente a Test.
2. Dar clic en el botón "Nuevo registro" (1).
3. Se abrirá la siguiente ventana [Clic aquí](#).
4. Llenar los datos correspondientes.
5. Dar clic en el botón Guardar.

Modificación de un registro:

1. Abrir el formulario correspondiente a Test.
2. Identificar el registro a modificar y dar clic en el botón "Ver" (2).
3. Se abrirá la siguiente ventana [Click aquí](#), con los datos a modificar.
4. Actualizar los datos correspondientes.
5. Dar clic en el botón Actualizar.

Eliminación de un registro:

1. Abrir el formulario correspondiente a Test.
2. Identificar el registro a eliminar y dar clic en el botón "Quitar" (3).
3. Se eliminará el registro seleccionado y se actualizará la tabla.

Consulta de un registro:

1. Abrir el formulario correspondiente a Test.
2. Escribir el criterio de búsqueda en la caja de texto (6).
3. Presionar la tecla Enter.
4. Si existen registros relacionados al criterio ingresado se presentaran en la tabla de los registros con las opciones para modificarlo o eliminarlo de ser el caso.

Activar una evaluación:

1. Abrir el formulario correspondiente a Test.
2. Dar clic en el botón "Activar" (4).

Agregar/Quitar rangos a una evaluación:

1. Abrir el formulario correspondiente a Test.
2. Dar clic en el botón "Agregar Rangos" (5).
3. Se abrirá la siguiente ventana [Click aquí](#), con las opciones para asignar/quitar los rangos de calificación.
4. Agregar/Quitar los datos correspondientes.
5. Dar clic en el botón Cerrar al concluir.

Agregar semestres a una evaluación:

1. Abrir el formulario correspondiente a Test.
2. Dar clic en el botón "Agregar Semestres" (6).
3. Se abrirá la siguiente ventana [Click aquí](#), con las opciones para

asignar/quitar los semestres.

4. Agregar/Quitar los datos correspondientes.

5. Dar clic en el botón Cerrar al concluir.

EVA-Soft | Administración

Inicio Areas Temas Preguntas **Test** Diseño Test Actuar Resultados Estadísticas Reportes

OPCIONES

Nuevo registro

DATOS:

NOMBRE	FECHA	TIEMPO	VALOR	PREGUNTAS	TIPO	CALIFICACION	ESTADO	OPCIONES
01 COMPRENSIÓN LECTURA	23/01/2014	60	800	70	GENERAL	CUALITATIVA	Disponible	Ver, Quitar, Activar, Agregar Energía, Agregar Semestres
T001	21/01/2014	15	100	10	GENERAL	CUALITATIVA	Disponible	Ver, Quitar, Activar, Agregar Energía, Agregar Semestres

Página 1

Ver

© 2013 Comité de Evaluación | Espam MFL

Pantalla que permite realizar la gestión de las evaluaciones.

1



Al dar clic, muestra una ventana emergente correspondiente a los campos necesarios para crear una evaluación.

2



Permite modificar el registro seleccionado

3



Permite eliminar el registro seleccionado

4



Permite activar la evaluación seleccionada, para que esté disponible desde el módulo de estudiantes.

5



Abre un cuadro de dialogo para configurar los rangos de calificación de la presente evaluación.

6



Abre un cuadro de dialogo para configurar los semestres que rendirán la presente evaluación.

7



Paginación de los registros

8

DATOS:									Pagina 1				
NOMBRE	FECHA	TIEMPO	VALOR	PREGUNTAS	TIPO	CALIFICACION	ESTADO	OPCIONES					
01 COMPRENSIÓN LECTURA	23/01/2014	60	800	70	GENERAL	CUALITATIVA	Disponible						
T001	21/01/2014	15	100	10	GENERAL	CUALITATIVA	Disponible						

Tabla contenedora de los registros existentes en el sistema.

Nuevo registro

EVA-Soft | Administración

ALFA LUMEN CANCELADO

NUEVO REGISTRO

Tipo: PARTICULAR 2

Nombre Test 3

Fecha dd/mm/aaaa 4

Numero preguntas 5

Calificación 6

Tipo calificación CUALITATIVA 7

Tiempo 8

Hora inicio --:-- 9

Estado PENDIENTE 10

Cancelar Guardar

Contenedor 1

11 12

OPCIONES

DATOS

NOMBRE	FECHA
01 COMPRENSIÓN LECTURA	23/01/2014
T001	21/01/2014

Página 1

1 Contenedor

NUEVO REGISTRO ✕

Tipo:	<input type="text" value="PARTICULAR"/>
Nombre Test	<input type="text"/>
Fecha	<input type="text" value="dd/mm/aaaa"/>
Numero preguntas	<input type="text"/>
Calificación	<input type="text"/>
Tipo calificación	<input type="text" value="CUALITATIVA"/>
Tiempo	<input type="text"/>
Hora Inicio	<input type="text" value="--:--"/>
Estado	<input type="text" value="PENDIENTE"/>

2

Control que permite seleccionar el ámbito de la evaluación a ingresar, particular para un grupo específico y general para todos y todas las estudiantes.

3

Control que permite ingresar el nombre o código de la evaluación.

4

Componente que permite ingresar la fecha de la evaluación.

5

Control que permite ingresar el número de preguntas total de la evaluación.

6

Control que permite ingresar la calificación total de la evaluación.

7

Control que permite escoger el método de calificación mostrado en los resultados.

8

Control que permite ingresar el tiempo de la evaluación en minutos.

9

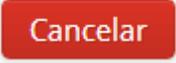
Control que permite ingresar la hora de inicio de la evaluación.

10

Estado

Opción que permite mantener disponible o no mostrar, la evaluación a ingresar ahora.

11

A red rectangular button with rounded corners and a white border, containing the word "Cancelar" in white text.

Botón que cierra la ventana emergente para nuevo registro.

12

A blue rectangular button with rounded corners and a white border, containing the word "Guardar" in white text.

Botón que guarda el registro ingresado en los elementos.

Configuración de Rangos

Datos:

Mínimo:

Máximo:

Descripción:

Agregados:

nr	RANGO	DESCRIPCIÓN	OPCIONES
1	0 - 399	INSATISFACTORIO	<input type="button" value="Quitar"/>
2	400 - 599	POCO SATISFACTORIO	<input type="button" value="Quitar"/>
3	600 - 699	SATISFACTORIO	<input type="button" value="Quitar"/>

1

Panel de configuración de los rangos de calificación correspondientes a una evaluación.

AGREGAR RANGOS ✕

Datos:

Minimo:

Maximo:

Descripción:

Agregar

Agregados

N°	RANGO	DESCRIPCIÓN	OPCIONES
1	0 - 399	INSATISFACTORIO	✕ Quitar
2	400 - 599	POCO SATISFACTORIO	✕ Quitar
3	600 - 699	SATISFACTORIO	✕ Quitar

Cerrar

2

Minimo:

Cuadro de texto para ingresar el rango inferior del intervalo.

3

Maximo:

Cuadro de texto para ingresar el rango superior del intervalo.

4

Descripción:

Cuadro de texto para ingresar la descripción del intervalo ingresado.

5

Agregar

Botón que guarda el intervalo ingresado, y refresca la tabla con los rangos asignados.

6

Agregados

Nº	RANGO	DESCRIPCIÓN	OPCIONES
1	0 - 399	INSATISFACTORIO	
2	400 - 599	POCO SATISFACTORIO	
3	600 - 699	SATISFACTORIO	

Tabla que va cargando los registro que se van agregando.

7



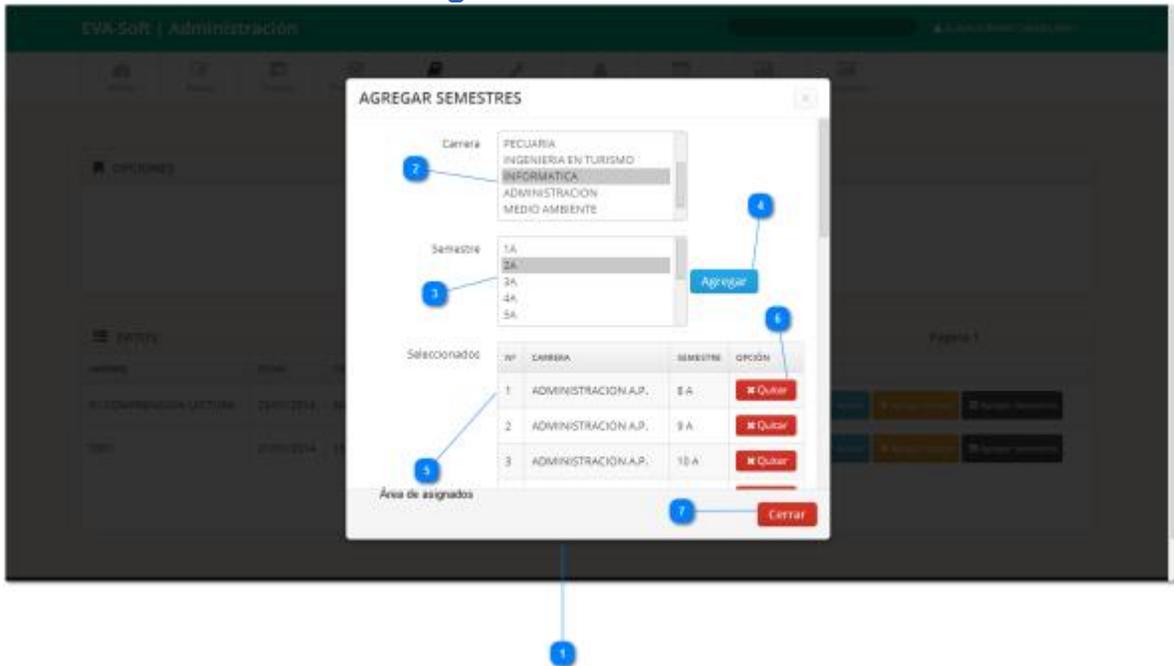
Botón que permite eliminar un registro ingresado.

8



Botón que cierra la ventana emergente.

Configuración de semestres



1

Área de registros de semestres

AGREGAR SEMESTRES ✕

Carrera

PECUARIA
 INGENIERIA EN TURISMO
INFORMATICA
 ADMINISTRACION
 MEDIO AMBIENTE

Semestre

1A
2A
 3A
 4A
 5A

Agregar

Seleccionados

Nº	CARRERA	SEMESTRE	OPCIÓN
1	ADMINISTRACION A.P.	8 A	✕ Quitar
2	ADMINISTRACION A.P.	9 A	✕ Quitar
3	ADMINISTRACION A.P.	10 A	✕ Quitar

Cerrar

2

Carrera

PECUARIA
 INGENIERIA EN TURISMO
INFORMATICA
 ADMINISTRACION
 MEDIO AMBIENTE

Componente que enlista las carreras correspondientes a la ESPAM MFL, al seleccionar alguna se filtran los semestres correspondientes.

3

Semestre

1A
2A
3A
4A
5A

Componente que enlista los semestres correspondientes a la carrera previamente seleccionada.

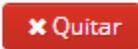
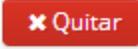
4

Agregar

Botón que agrega a la presente evaluación el semestre seleccionado.

5

Área de asignados

Seleccionados	Nº	CARRERA	SEMESTRE	OPCIÓN
	1	ADMINISTRACION A.P.	8 A	
	2	ADMINISTRACION A.P.	9 A	
	3	ADMINISTRACION A.P.	10 A	

6



Botón que permite eliminar el registro seleccionado.

7



Botón que permite cerrar la ventana.

Gestión de diseño de un Test

Diseñar una evaluación:

1. Abrir el formulario correspondiente a "Diseño Test".
2. Dar clic en el botón "Diseñar" (2).
3. Se mostrará la siguiente ventana [Clic aquí](#).
4. Llenar los datos correspondientes.
5. Dar clic en el botón Guardar.

Modificación de un registro:

1. Abrir el formulario correspondiente a Test.
2. Identificar el registro a modificar y dar clic en el botón "Ver" (2).
3. Se abrirá la siguiente ventana [Click aquí](#), con los datos a modificar.
4. Actualizar los datos correspondientes.
5. Dar clic en el botón Actualizar.

Eliminación de un registro:

1. Abrir el formulario correspondiente a Test.
2. Identificar el registro a eliminar y dar clic en el botón "Quitar" (3).
3. Se eliminará el registro seleccionado y se actualizará la tabla.

Consulta de un registro:

1. Abrir el formulario correspondiente a Test.
2. Escribir el criterio de búsqueda en la caja de texto (6).
3. Presionar la tecla Enter.
4. Si existen registros relacionados al criterio ingresado se presentaran en la tabla de los registros con las opciones para modificarlo o eliminarlo de ser el caso.

Activar una evaluación:

1. Abrir el formulario correspondiente a Test.
2. Dar clic en el botón "Activar" (4).

Agregar/Quitar rangos a una evaluación:

1. Abrir el formulario correspondiente a Test.
2. Dar clic en el botón "Agregar Rangos" (5).
3. Se abrirá la siguiente ventana [Click aquí](#), con las opciones para asignar/quitar los rangos de calificación.
4. Agregar/Quitar los datos correspondientes.
5. Dar clic en el botón Cerrar al concluir.

Agregar semestres a una evaluación:

1. Abrir el formulario correspondiente a Test.
2. Dar clic en el botón "Agregar Semestres" (6).
3. Se abrirá la siguiente ventana [Click aquí](#), con las opciones para

asignar/quitar los semestres.

4. Agregar/Quitar los datos correspondientes.

5. Dar clic en el botón Cerrar al concluir.

EVA-Soft | Administración

Inicio Areas Temas Preguntas Test **Diseño Test** Activar Resultados Estadísticas Reportes

ALANA ELMENY CAÑELARIO

EVALUACIONES

NOMBRE	FECHA	TIEMPO	VALOR	PREGUNTAS	TIPO	CALIFICACION	ESTADO	OPCIONES
T001	21/01/2014	15	100	10	GENERAL	CUALITATIVA	Disponible	<input type="button" value="Diseñar"/> <input type="button" value="Ver"/>
01 COMPRENSIÓN LECTURA	23/01/2014	60	800	70	GENERAL	CUALITATIVA	Disponible	<input type="button" value="Diseñar"/> <input type="button" value="Ver"/>

© 2013 Comité de Evaluación | Equim NPL

localhost:12485/ViewDesignTest.aspx#

1

Área de registros

EVALUACIONES									
NOMBRE	FECHA	TIEMPO	VALOR	PREGUNTAS	TIPO	CALIFICACION	ESTADO	OPCIONES	
T001	21/01/2014	15	100	10	GENERAL	CUALITATIVA	Disponible	<input type="button" value="Diseñar"/>	<input type="button" value="Ver"/>
01 COMPRENSIÓN LECTURA	23/01/2014	60	800	70	GENERAL	CUALITATIVA	Disponible	<input type="button" value="Diseñar"/>	<input type="button" value="Ver"/>

Tabla que presenta los elementos existentes en el sistema.

2

Diseñar

Botón que permite comenzar a configurar el presente registro.

3

Ver

Botón que permite visualizar las configuraciones ya asignadas a el registro

seleccionado.

Asignación de Áreas

DISEÑO EVALUACIÓN

AREAS

Materias: PROYECTO DE TESIS, DESARROLLO DE TESIS, ADMINISTRACION EMPRESARIAL, PRACTICA EMPRESARIAL

SubÁreas: [Empty list]

Porcentaje: Ingrese %

Agregar

AGREGADAS

ASIGNATURA	AREA	PORCENTAJE ACTUAL	CAMBIAR	OPCIONES
COMPRESION LECTORA Y ESCRITA	COMPRESION LECTORA	50	[Input field]	[Quitar] [Agregar preguntas]
COMPRESION LECTORA Y ESCRITA	COMPRESION ESCRITA	50	[Input field]	[Quitar] [Agregar preguntas]
Total %:		100		

1

Materias

PROYECTO DE TESIS
DESARROLLO DE TESIS
ADMINISTRACION EMPRESARIAL
PRACTICA EMPRESARIAL

Componente que enlista las asignaturas existentes en el sistema

2

SubAreas

[Empty list]

Componente que enlista las áreas relacionadas a una asignatura escogida

3

Porcentaje

Ingrese %

Control que permite ingresar el porcentaje que valdrá el área a configurar en la evaluación

4

Agregar

Agrega el registro ingresado

5

AGREGADAS				
ASIGNATURA	AREA	PORCENTAJE ACTUAL	CAMBIAR	OPCIONES
COMPRESION LECTORA Y ESCRITA	COMPRESION LECTORA	50	<input type="text"/>	<input type="button" value="✕ Quitar"/> <input type="button" value="✓ Agregar preguntas"/>
COMPRESION LECTORA Y ESCRITA	COMPRESION ESCRITA	50	<input type="text"/>	<input type="button" value="✕ Quitar"/> <input type="button" value="✓ Agregar preguntas"/>
Total %:		100		

Tabla que muestra las áreas asignadas a la evaluación

6

Control que permite cambiar el porcentaje agregado

7

Elimina el registro seleccionado

8

Permite abrir el panel correspondiente a los temas relacionados al área seleccionada.

9

Componente que muestra la sumatoria total de los porcentajes agregados, este valor no puede exceder a 100

Asignación de Temas

The screenshot shows a web interface for assigning topics. At the top, there's a 'TEMAS' section with a list of topics (1), a 'Porcentaje' input field (2), and a 'Cantidad Preguntas' input field (3). A blue button 'Agregar Temas' (4) is next to the percentage field. Below this is a table titled 'AGREGADOS' (5) with columns for AREA, TEMA, PROCENTAJE, CAMBIAR PROCENTAJE, PREGUNTAS, CAMBIAR PREGUNTAS, and OPCIONES. The table contains six rows of data. At the bottom, there are 'Total No:' (7) and 'Total Preguntas:' (8) fields.

AREA	TEMA	PROCENTAJE	CAMBIAR PROCENTAJE	PREGUNTAS	CAMBIAR PREGUNTAS	OPCIONES
COMPRESION ESCRITA	ORTOGRAFÍA	30	<input type="text"/>	15	<input type="text"/>	<input type="button" value="Quitar"/>
COMPRESION ESCRITA	SINÓNIMOS	20	<input type="text"/>	10	<input type="text"/>	<input type="button" value="Quitar"/>
COMPRESION ESCRITA	ANTÓNIMOS	20	<input type="text"/>	10	<input type="text"/>	<input type="button" value="Quitar"/>
COMPRESION ESCRITA	PALABRAS POLISÉMICAS	10	<input type="text"/>	4	<input type="text"/>	<input type="button" value="Quitar"/>
COMPRESION ESCRITA	SEMÁNTICA	10	<input type="text"/>	4	<input type="text"/>	<input type="button" value="Quitar"/>
COMPRESION ESCRITA	ORDENAR FRASES	10	<input type="text"/>	2	<input type="text"/>	<input type="button" value="Quitar"/>
Total No:		100		Total Preguntas:	45	

1

Temas

- ORTOGRAFÍA
- SINÓNIMOS
- ANTÓNIMOS
- PALABRAS POLISÉMICAS

Control que muestra los temas relacionados al área que seleccionamos.

2

Porcentaje

Control que permite ingresar el porcentaje correspondiente al tema a ingresar.

3

Cantidad Preguntas

Control que permite ingresar la cantidad de preguntas del tema a ingresar.

4

Agregar Temas

Agrega los elementos ingresados a la base de datos.

5

AGREGADOS						
AREA	TEMA	PROCENTAJE	CAMBIAR PROCENTAJE	PREGUNTAS	CAMBIAR PREGUNTAS	OPCIONES
COMPRESION ESCRITA	ORTOGRAFÍA	30	<input type="text"/>	15	<input type="text"/>	<input type="button" value="✖ Quitar"/>
COMPRESION ESCRITA	SINÓNIMOS	20	<input type="text"/>	10	<input type="text"/>	<input type="button" value="✖ Quitar"/>
COMPRESION ESCRITA	ANTÓNIMOS	20	<input type="text"/>	10	<input type="text"/>	<input type="button" value="✖ Quitar"/>
COMPRESION ESCRITA	PALABRAS POLISÉMICAS	10	<input type="text"/>	4	<input type="text"/>	<input type="button" value="✖ Quitar"/>
COMPRESION ESCRITA	SEMÁNTICA	10	<input type="text"/>	4	<input type="text"/>	<input type="button" value="✖ Quitar"/>
COMPRESION ESCRITA	ORDENAR FRASES	10	<input type="text"/>	2	<input type="text"/>	<input type="button" value="✖ Quitar"/>
Total %:		<input type="text" value="100"/>		Total Preguntas:	<input type="text" value="45"/>	

Tabla que enlista los elementos ingresados relacionados al área seleccionada.

6

✖ Quitar

Botón que permite eliminar el registro

7

100

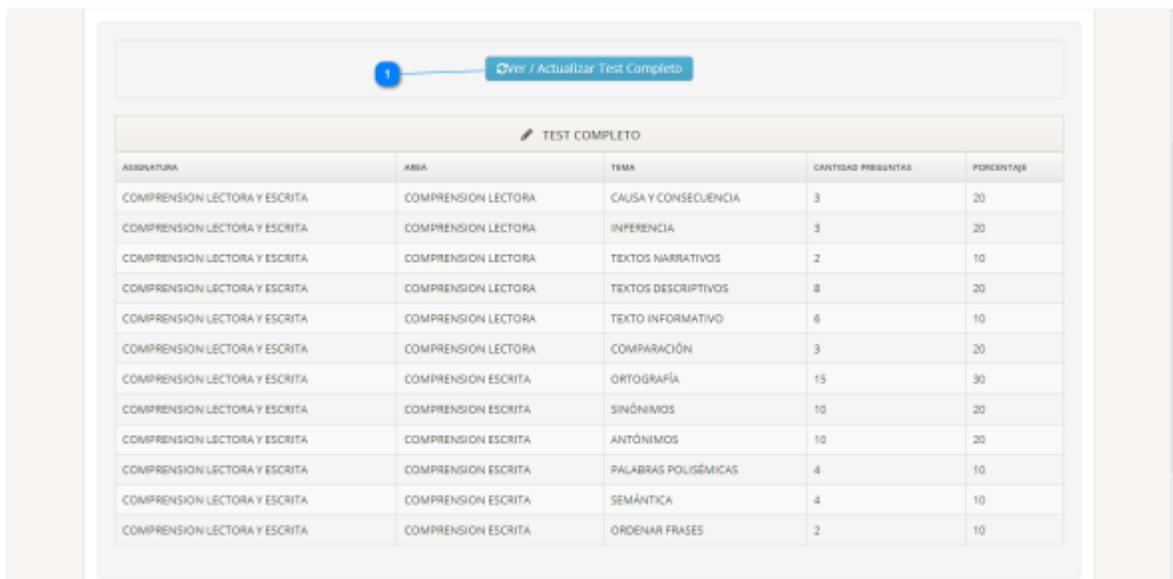
Control que muestra la suma del porcentaje asignado

8

45

Control que muestra la suma de la cantidad de preguntas asignadas

Visualización de test



The screenshot shows a web interface for viewing test results. At the top, there is a button labeled "Ver / Actualizar Test Completo" with a refresh icon. Below this is a table titled "TEST COMPLETO" with the following columns: ASIGNATURA, AREA, TEMA, CANTIDAD PREGUNTAS, and PORCENTAJE. The table contains 13 rows of data.

ASIGNATURA	AREA	TEMA	CANTIDAD PREGUNTAS	PORCENTAJE
COMPRESION LECTORA Y ESCRITA	COMPRESION LECTORA	CAUSA Y CONSECUENCIA	3	20
COMPRESION LECTORA Y ESCRITA	COMPRESION LECTORA	INFERENCIA	3	20
COMPRESION LECTORA Y ESCRITA	COMPRESION LECTORA	TEXTOS NARRATIVOS	2	10
COMPRESION LECTORA Y ESCRITA	COMPRESION LECTORA	TEXTOS DESCRIPTIVOS	8	20
COMPRESION LECTORA Y ESCRITA	COMPRESION LECTORA	TEXTO INFORMATIVO	6	10
COMPRESION LECTORA Y ESCRITA	COMPRESION LECTORA	COMPARACIÓN	3	20
COMPRESION LECTORA Y ESCRITA	COMPRESION ESCRITA	ORTOGRAFÍA	15	30
COMPRESION LECTORA Y ESCRITA	COMPRESION ESCRITA	SINÓNIMOS	10	20
COMPRESION LECTORA Y ESCRITA	COMPRESION ESCRITA	ANTÓNIMOS	10	20
COMPRESION LECTORA Y ESCRITA	COMPRESION ESCRITA	PALABRAS POLISÉMICAS	4	10
COMPRESION LECTORA Y ESCRITA	COMPRESION ESCRITA	SEMÁNTICA	4	10
COMPRESION LECTORA Y ESCRITA	COMPRESION ESCRITA	ORDENAR FRASES	2	10

1

Ver / Actualizar Test Completo

Botón que actualiza o presenta los registros asignados a una evaluación.

Módulo de activación de estudiantes

1

2

3

4

© 2013 Comité de Evaluación | Espam MFL

1

EVALUACIONES									
NOMBRE	FECHA	TIEMPO	VALOR	PREGUNTAS	TIPO	CALIFICACION	ESTADO	OPCIONES	
T001	21/01/2014	15	100	10	GENERAL	CUALITATIVA	Disponible	<input checked="" type="checkbox"/> ACTIVAR	
01 COMPRENSIÓN LECTURA	23/01/2014	60	800	70	GENERAL	CUALITATIVA	Disponible	<input checked="" type="checkbox"/> ACTIVAR	

Enlista las evaluaciones ingresadas en el sistema para seleccionar

2

ACTIVAR

Botón que permite visualizar los estudiantes validados para la evaluación seleccionada

3

ACTIVAR ESTUDIANTES					
Nº	NOMBRE	CARRERA	SEMESTRE	ESTADO	OPCIONES
1	ALCIVAR SANTANDER ADRIAN HERACLITO	INFORMATICA	10 A	nuñ	<input type="checkbox"/> Activar

Área que contiene la tabla que enlista y permite activar los estudiantes validados en la evaluación

4

 Activar

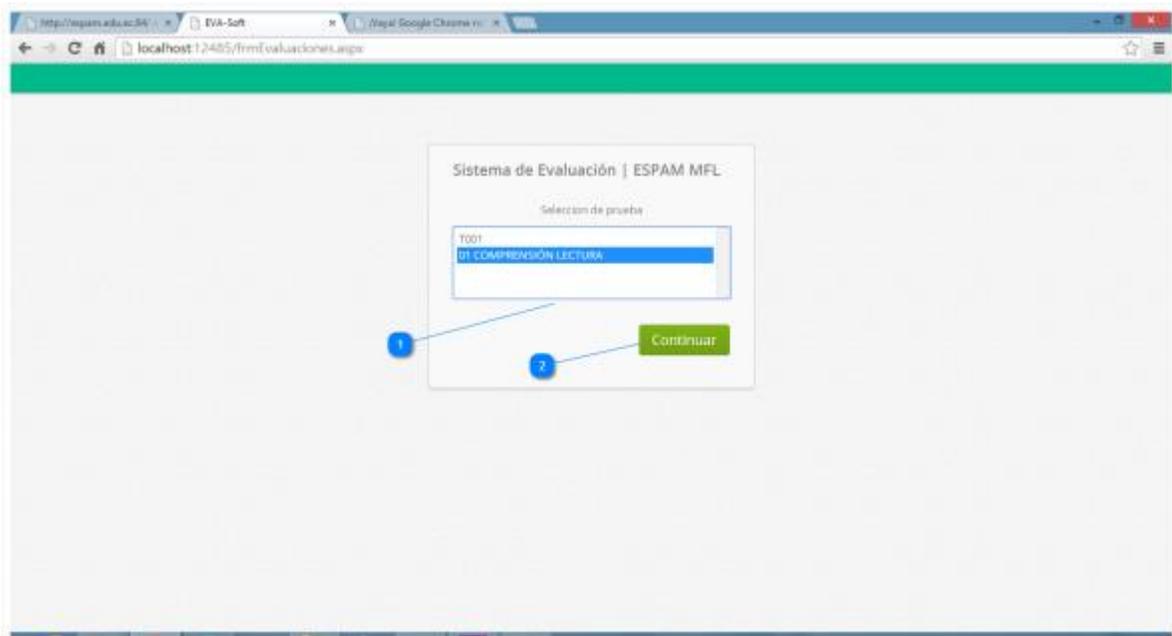
Control que permite activar individualmente a los estudiantes

5

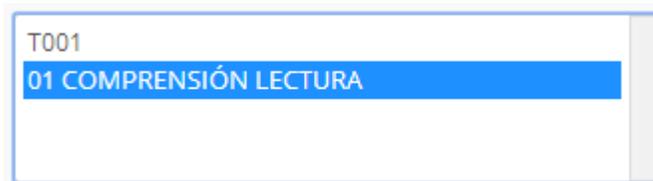
Activar Todos

Botón que permite activar a todos los estudiantes seleccionados

Módulo de evaluaciones



1



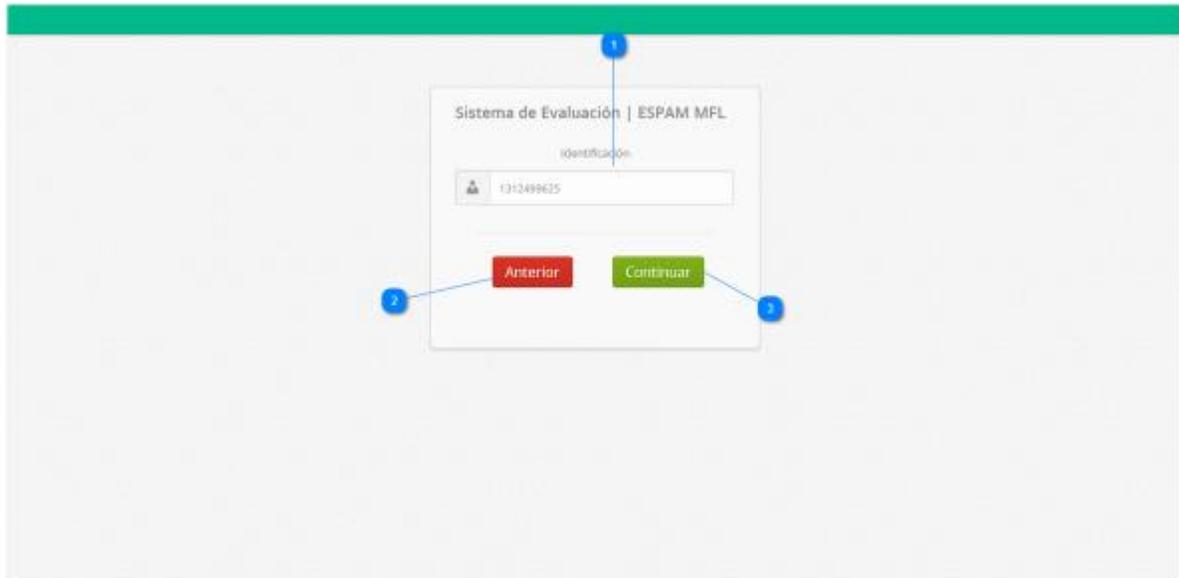
Control que muestra las evaluaciones programadas para la fecha actual

2

Continuar

Botón que permite avanzar al proceso de validación

Validación de estudiantes



1

Control que permite ingresar el número de cedula del estudiante

2

Anterior

Botón que permite regresar al módulo de selección de evaluación

3

Continuar

Botón que luego de confirmar la evaluación abre una ventana emergente con la evaluación

Pantalla de Evaluación

The screenshot shows a web-based evaluation interface. At the top, there is a header area (1) containing a logo on the left and student information on the right: Carrera: INFORMÁTICA, Cedula: 1312499625, Nombre: ALCIVAR SANTANDER ADRIAN HERACLITO, Semestre: 10 A, Código Evaluación: 01 COMPRENSIÓN LECTURA, and Tiempo: 60. Below the header is a green bar (2) indicating 'Pregunta 1 de 70' and containing 'Mostrar tiempo' (3) and 'Terminar' (4) buttons. The main content area (5) asks the user to 'Escoge el término genérico de cada ser, objeto o fenómeno' and provides a question: 'Un cuchillo es:'. Below this are four radio button options (6): A. Una navaja, B. Un instrumento, C. Un arma, and D. Ninguna de las anteriores. At the bottom of the question area are 'Anterior' and 'Siguiente' navigation buttons (7). The Windows taskbar is visible at the very bottom.

1

This block shows a close-up of the header area (1) from the evaluation interface. It includes the logo on the left and the student information on the right: Carrera: INFORMÁTICA, Cedula: 1312499625, Nombre: ALCIVAR SANTANDER ADRIAN HERACLITO, Semestre: 10 A, Código Evaluación: 01 COMPRENSIÓN LECTURA, and Tiempo: 60.

Área de información general sobre la evaluación

2

This block shows a close-up of the question bar (2) from the evaluation interface. It displays 'Pregunta 1 de 70' and contains the 'Mostrar tiempo' (3) and 'Terminar' (4) buttons.

Área de número de preguntas

3

This block shows a close-up of the 'Mostrar tiempo' (3) button from the evaluation interface. The button is green with a white eye icon and the text 'Mostrar tiempo'.

Botón que muestra o oculta el tiempo

4

A red rectangular button with a white right-pointing arrow icon and the text "Terminar" in white.

Botón para terminar la evaluación

5

Escoge el término genérico de cada ser, objeto o fenómeno.

Un cuchillo es:

Área de pregunta

6

Opciones

A Una navaja.

B Un instrumento.

C Un arma.

D Ninguna de las anteriores.

Área que muestra las opciones de respuesta

7

A blue rectangular button with a white left-pointing arrow icon and the text "Anterior" in white.

Botón para retroceder a la pregunta anterior

8

A blue rectangular button with the text "Siguiete" in white and a white right-pointing arrow icon.

Botón para continuar con la pregunta siguiente

Módulo de resultados generales

1

EVALUACIONES								
NOMBRE	FECHA	TIEMPO	VALOR	PREGUNTAS	TIPO	CALIFICACION	ESTADO	OPCIONES
T001	21/01/2014	15	100	10	GENERAL	CUALITATIVA	Disponible	Ver Resultados
01 COMPRENSIÓN LECTURA	07/02/2014	60	800	70	GENERAL	CUALITATIVA	Disponible	Ver Resultados

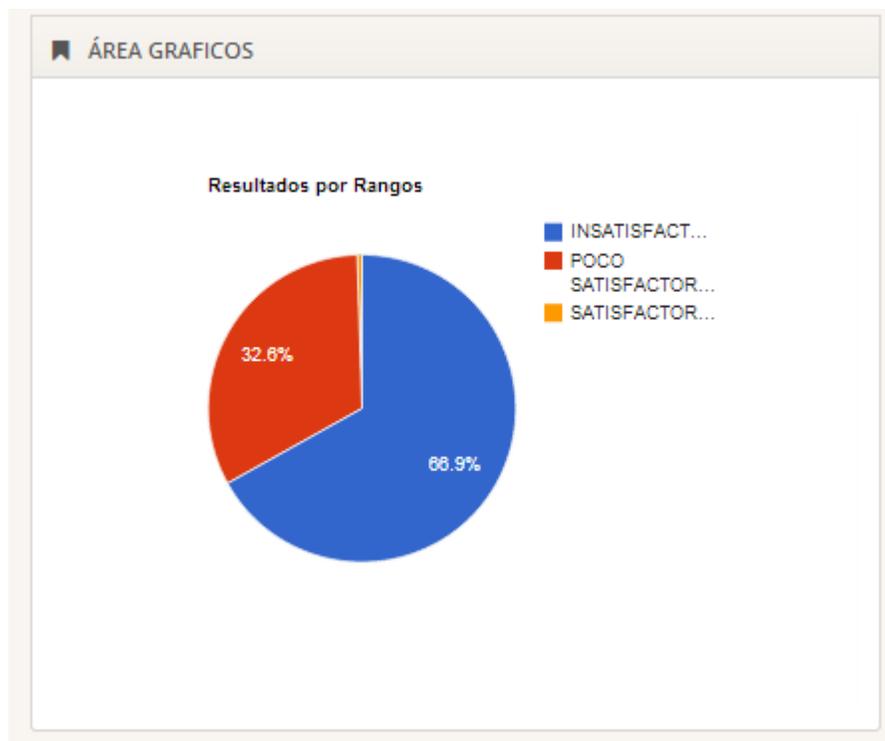
Utilice esta sección para seleccionar la evaluación de la cual necesita ver los resultados

2

[Ver Resultados](#)

Botón que mostrará los resultados de la evaluación seleccionada

3



Área que muestra los resultados por rangos configurados para la evaluación seleccionada.

4



Área que muestra los resultados por carreras configuradas para la evaluación seleccionada.

ÁREA INFORMACIÓN

RANGO	DESORPOCIÓN
0 - 399	INSATISFACTORIO
400 - 599	POCO SATISFACTORIO
600 - 699	SATISFACTORIO
700 - 800	SOBRESALIENTE

VER / OCULTAR TABLA

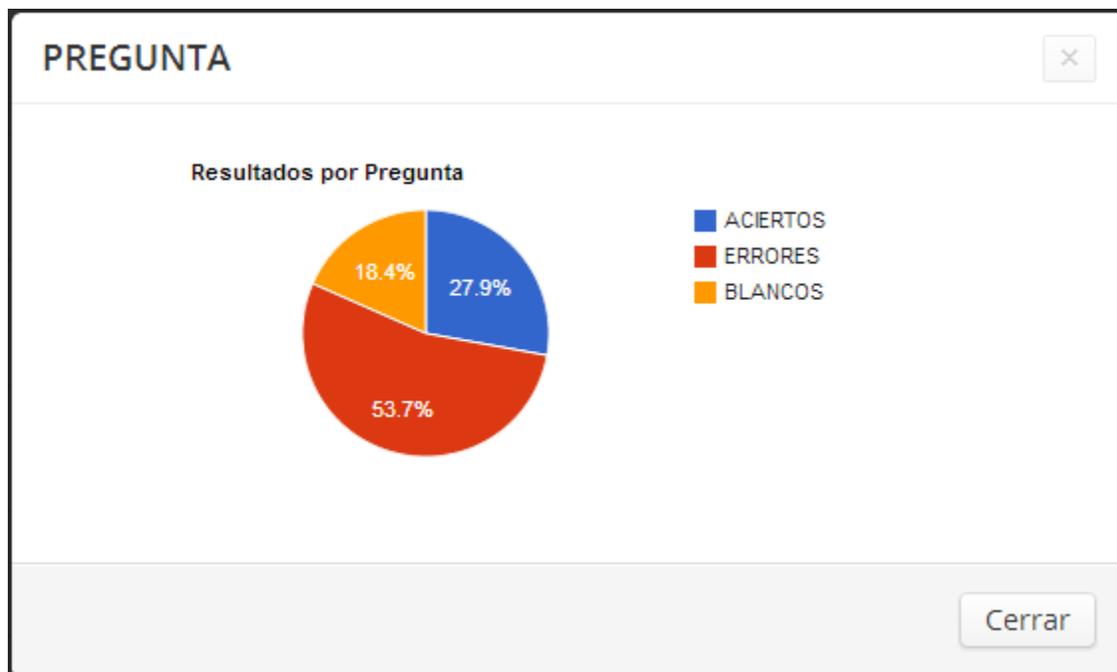
NOMBRE	ACERTOS	PENALIZADOS	ERRORES	BLANCOS	EQUIVALENTE	CALIFICACIÓN	Opciones
VERA REYES CARLA ALEXANDRA	57	51	13	0	SATISFACTORIO	653.02	Ver Detalles
SHIGLIANO PARRAGA ROBERTO CARLOS	52	43	18	0	SATISFACTORIO	611.02	Ver Detalles
BARREIRO MOLINA ARELY JULIANA	54	47	14	2	POCO SATISFACTORIO	593.26	Ver Detalles
PONCE ROMERO IRVING ABEL	49	39	20	1	POCO SATISFACTORIO	579.02	Ver Detalles
PINCAJ ORMAZA GEMA ALEJANDRA	53	45	17	0	POCO SATISFACTORIO	569.04	Ver Detalles
VILLAO LOOR MARIA BELEN	49	41	17	4	POCO SATISFACTORIO	568.6	Ver Detalles
MOREIRA PARRAGA GEMA JENIFFER	46	34	24	0	POCO SATISFACTORIO	564.56	Ver Detalles

Área que muestra los rangos y las mayores calificaciones de la evaluación seleccionada

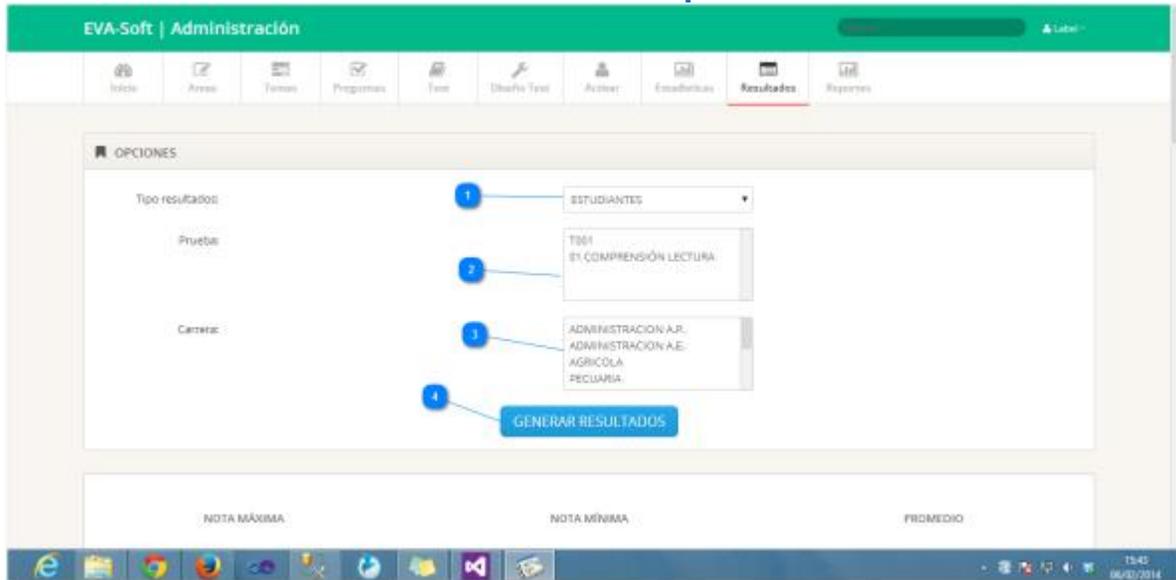
VER / OCULTAR PREGUNTAS

Nº	PREGUNTA	ACERTOS	ERRORES	BLANCOS	Opciones
1	Tomando en consideración el texto, ¿Qué significa la palabra extinción?	100	258	44	Ver Gráfico
2	Indique la palabra que está incorrectamente escrita.	157	180	56	Ver Gráfico
3	El empresario.....que los beneficios económicos no correspondían con sus presupuestos.	314	66	22	Ver Gráfico
4	Desafonar:	48	243	111	Ver Gráfico
5	ESPARCIMIENTO	196	180	26	Ver Gráfico
6	Una cuchara es:	321	62	19	Ver Gráfico
7	Una casa es:	187	192	23	Ver Gráfico
8	SUEÑO	67	303	32	Ver Gráfico
9	APRIORISTICO	150	94	158	Ver Gráfico

Área que muestra los resultados por preguntas, y al dar clic en el botón de la parte derecha "Ver gráfico" muestra un gráfico correspondiente a los resultados de la pregunta seleccionada.



Módulo de resultados por carrera



1

ESTUDIANTES

Control que permite seleccionar el tipo de resultados a visualizar, DOCENTES O ESTUDIANTES.

2

T001
01 COMPRENSIÓN LECTURA

Área de seleccion de la evaluación a visualizar resultados.

3

ADMINISTRACION A.P.
ADMINISTRACION A.E.
AGRICOLA
PECUARIA

Área de seleccion de la carrera a visualizar resultados.

4

GENERAR RESULTADOS

Botón que permite generar los resultados correspondientes a la evaluación y carrera seleccionada.

