

DESARROLLO DE UN SISTEMA DE INFORMACIÓN PARA LA TOMA DE DECISIONES
DE LA INSTITUCIÓN EDUCATIVA PRIMAVERA

CRISTHIAN CAMILO CORDOBA 0858903

JONNATAN ALBEIRO SÁNCHEZ 0858937

Universidad del Valle
Sede Tuluá
Programa Académico de Ingeniería en Sistemas
Tuluá, Mayo de 2013

DESARROLLO DE UN SISTEMA DE INFORMACIÓN PARA LA TOMA DE
DECISIONES DE LA INSTITUCIÓN EDUCATIVA PRIMAVERA

CRISTHIAN CAMILO CORDOBA

CRISTHIAN CAMILO CORDOBA

Trabajo de grado para optar por el título de Ingeniero de Sistemas

Director

Julián Andrés Rodas Laverde, Ing.

Universidad del Valle

Sede Tuluá

Programa Académico de Ingeniería en Sistemas

Tuluá, Mayo de 2013

NOTA DE ACEPTACIÓN

Presidente del jurado

Jurado

Jurado

DEDICATORIA

JONNATAN SÁNCHEZ

A mi madre con todo amor,
Mi hermano, mi padre,
Todos aquellos
Que me apoyaron siempre.

DEDICATORIA CRISTHIAN CORDOBA

A mi familia, a mis profesores y
A todos los que estuvieron
Con migo en este proceso.

AGRADECIMIENTOS

Jonnatan expresa su agradecimiento a:

A Dios por brindarme la sabiduría e inteligencia para superar este reto. A mi familia por apoyarme en este sueño y entender el dejar compartir tiempo con ellos, A mi novia por ayudarme y acompañarme durante todo este proceso.

Cristhian expresa su agradecimiento a:

A los profesores por ayudarme a obtener el conocimiento que tengo ahora, a mi familia toda por comprenderme y ayudarme, a mi madre que es la artífice de todo este sueño.

RESUMEN

Las pruebas virtuales se han convertido en una herramienta de gran importancia en los procesos de los centros educativos, enfocándola a mejorar la calidad de la educación al permitir a dichos centros medir de forma meticulosa, exacta, continua y rápida sus avances y sus falencias. De igual manera se pueden nombrar a las técnicas de minería de datos y datawarehousing, como adelantos informáticos enfocados a la inteligencia de negocios y que últimamente se han convertido en la mano derecha de muchas organizaciones en la búsqueda continua de ganar terreno en el mercado y en la toma de decisiones estratégicas de gran relevancia.

El proyecto fue enfocado en la implementación de una aplicación para la realización de pruebas tipo ICFES virtuales que proporcionará un entorno de exámenes difusible, rápido y con la capacidad de soportar diferentes tipos de pruebas mediante un banco de preguntas dinámico. Además de las gráficas estadísticas y reportes que se visualizan a partir de la aplicación, para el análisis y toma de decisiones basadas en los resultados de estas. Se hizo uso de un proceso de minería de datos contrastado con los resultados de las calificaciones de los estudiantes en las diferentes áreas en los períodos académicos, esto con el fin de tener una base de conocimiento que permita realizar un análisis del comportamiento de resultados en dichos rubros y así aportar a la toma de decisiones en búsqueda de la mejora de la calidad institucional.

Palabras clave: Pruebas virtuales pre-ICFES, DataMart, Inteligencia de negocios y Calidad Institucional.

ABSTRACT

Virtual tests have become a very important tool in the processes of schools by focusing on improving the quality of education to allow these centers meticulously measured, accurate, continuous and rapid progress and its shortcomings. Similarly you can name the data mining techniques and advances in computing datawarehousing as focused on business intelligence and lately have become the right hand of many organizations in the continued to gain ground in the market and in the strategic decisions of great importance.

The project focuses on the implementation of an application for type testing virtual ICFES provide a test environment diffusible, fast and with the ability to support different types of tests using a dynamic question bank. Besides the statistical graphs and reports are displayed from the application, for the analysis and decision making based on the results of these tests will make use of a data mining process contrasted with the results of student grades in different areas in academic terms, this in order to have a knowledge base that allows behavioral analysis of results in these areas and thus contribute to decision-making in pursuit of improving institutional quality.

Keywords: Virtual Testing pre-ICFES, DataMart, Business Intelligence and Institutional Quality.

Tabla de contenido

CAPITULO I	12
1. DESCRIPCIÓN GENERAL DEL PROBLEMA.....	12
1.1 Planteamiento del problema.....	12
1.2 Formulación del problema	13
2. JUSTIFICACIÓN DEL PROBLEMA	14
2.1 Justificación social	14
2.2 Justificación académica.....	14
3. OBJETIVOS	15
3.1 Objetivo general	15
3.2 Objetivos específicos.....	15
4. MARCO REFERENCIAL	16
4.1 Marco teórico	16
4.2 Diferencia entre metodologías tradicional y ágil.....	20
4.3 Principales metodologías tradicionales	20
4.4 Metodología ágil	22
4.5 Principales métodos ágiles	22
5. Estado del arte	25
CAPITULO II	26
6. Análisis.....	26
6.1 Metodología	26
6.2 Modelo de dominio	27
6.3 Modelo de prototipos	29
6.4 Diagrama de paquetes	33
7. Análisis y diseño preliminar.....	34
7.1 Casos de uso	34
7.2 Modelo de robustez	36
8. Diseño	38
8.1 Diagrama de secuencia.....	38
8.2 Esquema de base de datos	40

9. Implementación.....	41
9.1 Diagrama de componentes	41
9.2 Documento de pruebas.....	41
9.2.1 Prueba Crear Profesor	42
(Tabla 7).....	42
CAPITULO III	43
10. PRUEBAS ACADEMICAS	43
10.1 Banco de datos	43
10.2 Configuración de las pruebas.	46
10.3 Reportes especiales (minería de datos)	47
10.4 Estadísticas	57
11 Implementación.....	58
12. CONCLUSIONES	59
13. TRABAJOS FUTUROS	60
14. BIBLIOGRAFÍA.....	61

LISTA DE TABLAS

1. Tabla 1: Contraste entre almacenamiento operacional y en un datawarehouse.....	18
2. Tabla 2: Ejemplo reglas de asociación	20
3. Tabla3: Diferencias entre metodologías tradicionales y metodologías ágiles.....	21
4. Tabla 4:Ejemplo casos de uso.....	37
5. Tabla 5:Descripción de dimensiones.....	43
6. Tabla 6: Descripción de hechos.....	48
7. Tabla 7: Ejemplo de tabla de pruebas.....	53

LISTA DE FIGURAS

1. Figura1:En esta figura se ven el ciclo de vida de la SSADM.....	24
2. Figura1: Modelo de dominio del sistema.....	31
3. Figura2: Primer ejemplo de prototipo del sistema.....	34
4. Figura3: Segundo ejemplo de prototipo del sistema.....	35
5. Figura4: Diagrama de paquetes del sistema.....	36
6. Figura5: Imágenes de guía para la creación de los diagramas de casos de uso.....	38
7. Figura6: Ejemplo de los diagramas de casos de uso.....	39
8. Figura7: Imágenes de guía para la creación de los diagramas de robustez.....	40
9. Figura8: Primer Ejemplo de un diagrama de robustez del sistema.....	40
10. Figura9: Segundo Ejemplo de un diagrama de robustez del sistema.....	41
11. Figura10: Primer Ejemplo de diagrama de secuencia del sistema.....	42
12. Figura11: Segundo Ejemplo de diagrama de secuencia del sistema.....	43
13. Figura12: Estrella de resultado por competencia.....	50
14. Figura13: Estrella de resultado por componente	51
15. Figura14: Estrella de resultado por área.....	52
16. Figura15:Diagrama de componentes del sistema.....	53

LISTA DE ANEXOS

Anexo A: Historias de Usuario del sistema.

Anexo B: Pruebas a todos los formularios y opciones del sistema.

Anexo C: Requerimientos del sistema.

Anexo D: Roles que pueden adquirir los usuarios de la aplicación.

Anexo E: Ejemplo carga de componentes y competencias desde Excel al sistema.

Anexo F: Ejemplo Cargar áreas desde Excel al sistema.

Anexo G: Ejemplo Cargar estudiantes desde Excel al sistema.

CAPITULO I

1. DESCRIPCIÓN GENERAL DEL PROBLEMA

1.1 Planteamiento del problema

Las Instituciones Educativas están obligadas a realizar procesos de auto-evaluación para llevar un informe de estos a la secretaría de educación, y de esta forma puedan ser catalogadas dentro de unas categorías, y de acuerdo a estas tener la posibilidad de participar por estímulos financieros.

Para que la Institución Educativa Primavera aumente su rendimiento académico y brinde un buen informe debe supervisar la calidad de las pruebas de auto-evaluación y para esto se debe llevar registro histórico y responder a preguntas que le permitan saber si hay un retroceso académico o no, y así puedan enfatizar en corregir sus debilidades.

Para los Centros Educativos es importante evaluar la capacidad de los estudiantes en pruebas internas de pre-ICFES, las cuales además permiten una mejor preparación de estos para las pruebas ICFES, que es una de las formas principales del gobierno para evaluar el desempeño de los centros de educación.

Realizar pre-ICFES en cada grado es una estrategia implementada por la Institución Educativa Primavera para mejorar su rendimiento académico, pero el principal problema de esta es el elevado costo que conlleva su realización, debido al volumen de los estudiantes que presentan estas pruebas y la gran cantidad de áreas que deben ser evaluadas. Los reducidos presupuestos que manejan estas instituciones les obligan a que las estrategias o planes de mejora de calidad tengan el menor costo posible.

La naturaleza física y la extensión de los exámenes pre-ICFES hace que se consuman una cantidad considerable de recursos y dinero de la institución, y adicional a esto los docentes deben destinar una gran cantidad de su tiempo en la labor muchas veces tediosa de revisión y calificación de estos.

A medida que pasa el tiempo la información existente de pre-ICFES se va a incrementar, ya que este proceso es un gran proveedor de documentos e información, como medida para contrarrestar este incremento desmesurado muchos datos históricos serán desechados por la institución, lo cual afectará considerablemente la calidad académica medida a través de los pre-ICFES, que perderá información de importancia para llevar un estudio adecuado de la evolución académica. Normalmente la información digitalizada en comparación a la cantidad de esta en forma física es más sustentable a medida que pasa el tiempo, lo que da un indicio de que la falta de sistematización de datos relevantes como los de pre-ICFES se presenta como un problema que está afectando directamente la institución educativa.

Las preguntas de calidad planteadas por la Institución Educativa Primavera tienen una base en los avances y retrocesos medidos a partir de los resultados de pre-ICFES y calificaciones académicas de los diferentes períodos lectivos, pero además de la antes nombrada pérdida de información relevante que puede afectar la correcta realización de estas evaluaciones de calidad, se carece de una herramienta que permita el tratamiento de grandes cantidades de datos, que permita responder

preguntas complejas de forma eficiente y con eficacia, y que muestre reportes importantes para la detección de fallas y puntos de interés para un mejoramiento del rendimiento académico. Esta información de carácter clave es comúnmente difícil de extraer sin una herramienta o una aplicación especializada.

A futuro, la Institución Educativa Primavera puede verse afectada financieramente y podría recortar de su estrategia de calidad procesos costosos como las pruebas pre-ICFES, requiriendo la implementación casi inmediata de herramientas como las tecnologías de la información que les permitan sustentar económicamente este tipo de procesos y que les hagan más fácil su gestión y realización, y que además permitan extraer información relevante para la toma de decisiones.

Si no se buscan soluciones a las problemáticas antes tratadas lo más seguro es que la institución educativa se estanque o no avance lo suficiente para lograr subir su categoría y poder aspirar a beneficios financieros por parte del gobierno que le permita crecer como institución.

1.2 Formulación del problema

¿Cómo aportar al mejoramiento de la eficiencia del proceso de pruebas académicas (pre-ICFES) y a un mejor análisis de los resultados de las mismas para cada estudiante en los diferentes periodos académicos en la Institución Educativa Primavera, con las herramientas, técnicas y metodologías que las ciencias de la computación ofrecen en la actualidad?

2. JUSTIFICACIÓN DEL PROBLEMA

2.1 Justificación social

Las tecnologías de la información cada vez toman más relevancia en las operaciones y procesos de las diferentes organizaciones, ya que ofrecen rapidez, control, confiabilidad y escalabilidad. Las aplicaciones de pre-icfes virtuales ofrecen la posibilidad a las instituciones educativas de controlar y de llevar a cabo con mayor eficacia, eficiencia los procesos de pruebas al interior de la institución. Anexo a esto las técnicas de datawarehousing ofrecen una oportunidad al interior de las organizaciones de responder ciertas preguntas que la apoyen en la toma de decisiones, y aplicado a las instituciones educativas pueden ser de gran ayuda para conocer las falencias en el proceso educativo mediante el estudio de los resultados de las pruebas pre-ICFES y las calificaciones académicas recopiladas a través de cierto lapso de tiempo, y de esta manera tomar decisiones que influyan en el mejoramiento de la calidad.

Los principales beneficiados con este proyecto son los estudiantes, profesores y directivos de la Institución Educativa Primavera, ya que:

- Permite a los estudiantes, un aprendizaje mucho más interactivo, al tener a su disposición una herramienta de las tecnologías de la información que hará las pruebas mucho más dinámicas, agradables e intuitivas.
- En cuanto al módulo de pre-ICFES, le permite a los profesores de la Institución guardar preguntas para su posterior reutilización, además la aplicación se encarga del procesamiento y calificación de dichas pruebas, previniendo errores en calificaciones y haciendo este proceso más dinámico, eficiente y rápido, lo que les da a los docentes un ahorro de tiempo destinado a estas labores que pueden llegar a ser dificultosas y extensas.
- Le proporciona a la parte directiva de la Institución una herramienta de apoyo en la evaluación del desempeño de los profesores y estudiantes de dicha institución, en pro de mejorar la calidad académica.

2.2 Justificación académica

La realización del proyecto, junto con las investigaciones que el desarrollo de este conllevó, sobre todo en la parte de datawarehousing, permitió adquirir nuevos conocimientos correspondientes al área del descubrimiento del conocimiento de las bases de datos, al igual que reforzó los conocimientos ya adquiridos en dicha área.

Además, reforzó los conocimientos y prácticas utilizadas en el momento de realizar el proceso de desarrollo de software, permitiendo analizar las metodologías más beneficiosas para realizar el tipo de proyecto trazado, y estudiar los diferentes aspectos que se deben tener en cuenta para aplicar correctamente la metodología escogida.

Adicional a esto, permitió adquirir conocimientos sobre los procesos que se llevan a cabo en un ambiente organizacional (Institución Educativa Primavera) y la forma en que interactúan entre sí para llevar a cabo resultados sostenibles y exitosos.

3. OBJETIVOS

3.1 Objetivo general

Desarrollar un sistema de información para la toma de decisiones del área de secretaría académica de la Institución Educativa Primavera.

3.2 Objetivos específicos

1. Desarrollar una aplicación web para la realización de pruebas virtuales pre-ICFES en la Institución Educativa Primavera que permita realizar y calificar dichas pruebas.
2. Implementar un módulo de consultas que permita visualizar información estadística-histórica de los resultados de las pruebas, a través de reportes estadísticos.
3. Diseño e Implementación de un data mart concerniente con el proceso de pruebas pre-ICFES.
4. Implementar la técnica de minería de datos asociación para la extracción de información que sirva como apoyo a la toma de decisiones.

4. MARCO REFERENCIAL

4.1 Marco teórico

El propósito de este trabajo fue el de diseñar e implementar una aplicación que permite realizar pruebas de pre-ICFES de manera virtual, además del diseño de un datamart, la implementación de un data mart y la realización de un proceso de minería de datos para la Institución Educativa Primavera, con el fin de lograr la optimización del manejo de la información y hacer un aporte a la toma de decisiones dentro de la Institución. Por lo anterior se concluye que las áreas del conocimiento que fueron utilizadas para el proyecto son:

- Ingeniería del software.
- Descubrimiento del conocimiento (KDD) (DataMart y Minería de Datos)
- Sistema de apoyo a la toma de decisiones (DSS).
- Metodologías de desarrollo (Agiles y tradicionales)

A continuación se nombraran algunos conceptos de los temas nombrados anteriormente.

4.1.1 Ingeniería de software.

4.1.1.1 Introducción

La Ingeniería de Software es una ciencia de la computación, que ofrece métodos y técnicas para desarrollar Software que soluciona todo tipo de problemas. Hoy día la Ingeniería del Software es reconocida como una nueva área de la ingeniería. El Ingeniero del Software esta cada vez más implantado como una profesión a nivel internacional, con derechos, deberes y responsabilidades que cumplir.¹

4.1.1.2 Ingeniería

La ingeniería es el estudio y la aplicación de cada una de las ramas de la tecnología. El profesional en este ámbito es conocido como ingeniero. El trabajo del ingeniero es idear la forma en la que una idea se convierta en realidad.²

4.1.1.3 Software

El software no son sólo programas, sino todos los documentos asociados y la configuración de datos que se necesitan para hacer que estos programas operen de manera correcta. Un sistema de software consiste en diversos programas independientes, archivos de configuración que se utilizan para ejecutar estos programas, un sistema de documentación que describe la estructura del sistema, la documentación para el usuario que explica cómo utilizar el sistema y sitios web que permitan a los usuarios descargar la información de productos recientes.

¹Ian Sommerville. **Ingeniería de Software**. Pearson. 2005

²[Std. 729, IEEE]

4.1.1.4 Ingeniería de software

La actividad del ingeniero supone la concreción de una idea en la realidad. Esto quiere decir que, a través de técnicas, diseños y modelos, y con el conocimiento proveniente de las ciencias, la ingeniería puede resolver problemas y satisfacer necesidades humanas.

La ingeniería también supone la aplicación de la inventiva y del ingenio para desarrollar una cierta actividad. Esto, por supuesto, no implica que no se utilice el método científico para llevar a cabo los planes.³

4.1.2. Descubrimiento del conocimiento (KDD)

Existen en el mundo real enormes masas de datos en las cuales hay una gran cantidad de información "oculta", de gran importancia estratégica, a la que no se puede acceder por las técnicas clásicas de recuperación de la información. El descubrimiento de esta información "oculta" es posible gracias a la Minería de datos, que se encarga de encontrar patrones y relaciones dentro de los datos permitiendo la creación de modelos, es decir, simulación de escenarios abstractos que se puedan dar en la realidad, pero es el descubrimiento del conocimiento que se encarga de la preparación de los datos y la interpretación de los resultados obtenidos, los cuales dan un significado a estos patrones encontrados.⁴

4.1.2.1 Data warehouse

4.1.2.1.1 Introducción

Es un repositorio de datos de muy fácil acceso, alimentado de numerosas fuentes, transformadas en grupos de información sobre temas específicos de negocios, para permitir nuevas consultas, análisis, reportes y decisiones.

El data warehouse, es actualmente, el centro de atención de las grandes instituciones, porque provee un ambiente para que las organizaciones hagan un mejor uso de la información que está siendo administrada por diversas aplicaciones operacionales.

Se puede caracterizar un data warehouse haciendo un contraste de cómo los datos de un negocio almacenados en un data warehouse, difieren de los datos operacionales usados por las aplicaciones de producción, como se ve claramente en la tabla 1.

³ Eric J. Braude. En el libro Ingeniería de Software: Una perspectiva orientada a objetos. Editorial Ra-Ma. 2003

^[4,6]Facultad de Ingeniería Industrial. Instituto de Investigación en el documento científico DATA MINING AND KNOWLEDGE DISCOVERY de la *Universidad Nacional Mayor de San Marcos, Lima Perú, 2012*

Base de Datos Operacional	Data Warehouse
Datos Operacionales	Datos del negocio para Información
Orientado a la aplicación	Orientado al sujeto
Actual	Actual + histórico
Detallada	Detallada + más resumida
Cambia continuamente	Estable

(Tabla 1)

Un data warehouse es un repositorio de datos que proporciona una visión global, común e integrada de los datos de la organización-independientemente de cómo se vayan a utilizar posteriormente por los consumidores o usuarios-, con las propiedades siguientes: estable, coherente, fiable y con información histórica. Al abarcar un ámbito global de la organización y con un amplio alcance histórico, el volumen de datos puede ser muy grande (centenas de terabytes)⁵.

4.1.2.2 Data mart

- “Un **data mart** cumple los mismos principios que un **data warehouse**, construir un repositorio de datos único, consistente, fiable y de fácil acceso.”

La diferencia de un data mart con un data warehouse es el alcance, mientras el data mart se piensa para cubrir las necesidades específicas de un departamento de la empresa. El data warehouse soluciona los problemas de una organización en su totalidad.

4.1.2.3 Data mining

- “La **data mining** es una etapa dentro del proceso completo del descubrimiento del conocimiento, este intenta obtener patrones o modelos a partir de los datos recopilados.”⁶

Es una poderosa tecnología que permite la extracción de información oculta de grandes bases de datos. Esta tecnología tiene un gran potencial ya que se está convirtiendo en una herramienta indispensable para las grandes compañías dándoles las herramientas para que se tomen decisiones de negocio.

4.1.2.4 Herramientas de data mining

Las herramientas de data mining pueden responder a preguntas de negocios que tradicionalmente consumen demasiado tiempo para poder ser resueltas y a los cuales los usuarios de esta información casi no están dispuestos a aceptar.

Estas herramientas exploran las bases de datos en busca de patrones ocultos, encontrando información predecible que un experto no puede llegar a encontrar porque se encuentra fuera de sus expectativas.

Las herramientas de data mining empleados en el proceso de KDD se pueden clasificar en dos grandes grupos:

⁵Jordi Conesa Caralat y Josep Curto Diaz en el libro Introducción al Business Intelligence, 14)

- Técnicas de verificación, en las que el sistema se limita a comprobar hipótesis suministradas por el usuario.
- Métodos de descubrimiento, en los que se han de encontrar patrones potencialmente interesantes de forma automática, incluyendo en este grupo todas las técnicas de predicción.

El resultado obtenido con la aplicación de algoritmos de data mining pertenecientes al segundo grupo, el de técnicas de descubrimiento, pueden ser de carácter descriptivo o predictivo. Las predicciones sirven para prever el comportamiento futuro de algún tipo de entidad mientras que una descripción puede ayudar a su comprensión.

4.1.4 Sistemas de Apoyo a las Decisiones

Se define a los sistemas de apoyo a las Decisiones como un conjunto de programas y herramientas que permiten obtener oportunamente la información requerida durante el proceso de la toma de decisiones, en un ambiente de incertidumbre.

Las principales características de estos sistemas son:

- Se introducen en la mayoría de los casos tras implantar sistemas que optimizan los procesos que almacenan grandes volúmenes de datos, pero no analizan los mismos, ya que estos constituyen su plataforma de información.
- La información que es generada por estos sistemas es de gran ayuda para todos los mandos de la empresa en el proceso de toma de decisiones.
- Este tipo de sistema no ahorra mano de obra. Por tanto justificar económicamente este tipo de sistemas es difícil, puesto que no se conocen los ingresos que traerá la inversión en el proyecto.
- Deben ser Sistemas de Información interactivos y amigables, con un excelente diseño visual, que se dirigen al usuario final.
- Apoyan la toma de decisiones que son naturalmente repetitivas pero con determinaciones no estructuradas.
- Estos sistemas pueden ser desarrollados directamente por el usuario final sin la participación operativa de los analistas y programadores del área de Informática.⁷

⁷ Leyba Cortez, Esteban y Prieto Tinoco, José en el libro Sistemas y Aplicaciones Informáticas en la página 77.
<http://fccea.unicauca.edu.co/old/tiposdesi.htm>

4.2 Diferencia entre metodologías tradicional y ágil

En la tabla 3 se ve la comparación entre las metodologías tradicionales y las ágiles.

"

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

“8

(Tabla 3)

⁹4.3 Principales metodologías tradicionales

Las principales metodologías son:

- Metodología MERISE
- Metodología SSADM (Análisis De Sistemas Estructurados y Métodos de Diseño)
- Metodología MÉTRICA

⁸ Téllez, Linda Luna metodología ágil vs metodología tradicional.: Universidad Gabriel Rene Romero. Facultad deIngeniería de Sistemas; 2012.

⁹ Leyba Cortez, Esteban y Prieto Tinoco, José en el libro Sistemas y Aplicaciones Informáticas en la página 77.

4.3.1 Metodología MERSIE:

La metodología MERSIE ha proporcionado un ciclo de vida más largo, ya que se escoge una etapa de planificación previa al desarrollo que se conoce como esquema director, además de la inclusión de dos ciclos más el de abstracción y el de la decisión; el de abstracción tiene tres niveles que son:

- **Nivel conceptual:** Finalidad del sistema.
- **Nivel organizativo:** Organización adecuada.
- **Nivel físico:** Se pregunta cuáles serán los medios técnicos.

Las fases de la metodología son:

1. Estudio preliminar: Estudia la situación que existe y plantea una solución total.

2. Estudio Detallado: Se examina lo que se va a desarrollar.

3. Implementación: Se realizan las aplicaciones o programas que se requieran, se divide en:

- Estudio Técnico
- Producción de programas

4. Realización y puesta en marcha

4.3.2 Metodología SSADM

Los aspectos importantes de la metodología SSDAM son:

- Se enfatiza en los usuarios, dándole importancia a los requisitos y la participación de los mismos.
 - Se define el proceso de producción, que se va a hacer, cuando se va a hacer y cómo se va a hacer.
 - Sus artefactos se hacen desde tres puntos de vista, los datos, los eventos y los procesos.
 - Proporciona la máxima flexibilidad en las herramientas que se utilizan y las técnicas de la implementación.
 - Esta metodología no cubre aspectos como la planificación o la construcción del código.
- 4.6.3 Metodología MÉTRICA

Esta metodología se enfoca directamente a al desarrollo del sistema. Se divide en cinco fases que son:

- Fase 0: Plan de sistemas de información.
- Fase 1: Análisis de sistemas.
- Fase 2: Diseño del sistema.
- Fase 3: Construcción del sistema.
- Fase 4: Implementación del sistema.

4.4 Metodología ágil

Una metodología ágil es toda aquella metodología que esboza los valores y principios que permiten a un equipo desarrollar software rápidamente y respondiendo a los cambios que se puedan presentar durante el desarrollo del proyecto.

4.4.1 Manifiesto ágil

La filosofía ágil se encuentra consignada en el manifiesto ágil, este documento surgió en febrero de 2001 en Utah-EEUU donde se creó también la “*The Agile Alliance*” una entidad sin ánimo de lucro dedicada a promover el uso de las metodologías ágiles en los procesos de desarrollo de software.

A continuación se presenta el manifiesto ágil este documento es de dominio popular:

- **“Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas.** La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.”
- **Desarrollar software que funciona más que conseguir una buena documentación.** La regla a seguir es “no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante”. Estos documentos deben ser cortos y centrarse en lo fundamental.
- **La colaboración con el cliente más que la negociación de un contrato.** Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.
- **Responder a los cambios más que seguir estrictamente un plan.** La habilidad de responder a los cambios que puedan surgir a lo largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.”¹⁰

4.5 Principales métodos ágiles

Los métodos ágiles más usadas son:

- SCRUM
- XP (Programación extrema)
- ICONIX

¹⁰Letelier Torres Patricio y Sánchez López Emilio A. en el libro Metodologías ágiles en el desarrollo de software,2

4.5.1 Metodología SCRUM

SCRUM es un proceso en el que se aplican los conceptos de buenas prácticas para trabajar de forma colaborativa, en equipo, con veras de obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección viene del estudio de cómo se debe trabajar en equipo.

En SCRUM se realizan entregas parciales y de una manera regular del producto final, siendo prioritarias aquellas entregas que le proporcionan mayor beneficio al cliente, por ello, SCRUM está diseñado para proyectos complejos, donde se necesita obtener resultados rápidos y que los requisitos son muy cambiantes o sin mucha definición, donde las bases primordiales son la innovación, la competitividad y la productividad.

Etapas del proceso SCRUM

- **Etapas de iniciación:** Se crea el primer piloto del sistema.
- **Etapas de lista de control del proyecto:** Se crea una lista que contenga todas las actividades que deben ser realizadas.
- **Etapas de iteración:** En esta etapa se contempla el rediseño y la implementación.

4.5.2 Metodología XP (programación extrema)

La programación extrema es una metodología de desarrollo ágil basada en una serie de valores y una docena de prácticas que son:

- ✓ **Equipo Complemento:** El equipo lo conforman todos aquellos que interfieren de alguna u otra manera con el proyecto, incluido el cliente y el responsable del proyecto.
- ✓ **Planificación:** Se realizan las historias de usuario y se documenta en qué orden se van a hacer las mini-versiones. La planificación debe ser revisado continuamente
- ✓ **Test del cliente:** El cliente, con ayuda de los desarrolladores del proyecto define las pruebas que se van a realizar para comprobar las mini-versiones.
- ✓ **Versiónes pequeñas:** Las mini-versiones tienen que ser tan pequeñas que puedan realizarse en pocas semanas, además deben de ser versiones que ofrezcan algo útil para el usuario no porciones de código que no se puedan ver funcionando.
- ✓ **Diseño simple:** Hacer siempre lo imprescindible y mantener el código lo más sencillo posible.
- ✓ **Pareja de programadores:** Los programadores se ubican de a dos por cada ordenador y se hacen cambio de parejas mínimo una vez por día.
- ✓ **Desarrollo guiado por las pruebas automáticas:** Se deben realizar programas que realicen pruebas automáticas a la aplicación y se deben ejecutar regularmente.
- ✓ **Integración continua:** Debe tenerse un ejecutable del proyecto que funcione y cada que se cree una nueva funcionalidad debe integrarse, recompilarse y probarse. No se debe esperar

mucho tiempo para realizar las integraciones ya que de ocurrir un error sería difícil saber dónde se encuentra el error.

- ✓ **El código es de todos:** Cualquiera de los integrantes del equipo de trabajo puede modificar cualquier parte del código y todo el grupo lo debe conocer totalmente.
- ✓ **Normas de codificación:** Todos los integrantes del grupo deben tener el mismo estilo de programación, que se vea que el código es de una sola persona.
- ✓ **Metáforas:** Hay que buscar unas frases o nombres que definan cómo funcionan las distintas partes del programa, de forma que sólo con los nombres se pueda uno hacer una idea de qué es lo que hace cada parte del programa.
- ✓ **Ritmo sostenible:** Debe definirse un ritmo y mantenerse, de modo que no hallan días en los que no se trabaje ni días en los que se trabaje en exceso, se debe tener claro las metas que se deben cumplir y luchar día a día para lograr la meta cercana.

4.5.3 ICONIX

ICONIX es una metodología que se encuentra en el medio entre RUP (Rational Unified Process) y un XP (eXtreme Programming).

ICONIX deriva directamente de RUP y en la mayoría de los casos solo necesita el 20% de los artefactos de UML, con lo cual se simplifica mucho el proceso de desarrollo, no se pierde documentación tan solo se deja lo necesario, esto implica que ICONIX da la oportunidad de trabajar dinámicamente con UML de tal forma que siempre se pueda utilizar artefactos distintos a los estipulados.

4.5.3.1 Análisis de requisitos en ICONIX.

En esta primera fase se crea un diagrama de dominio que no es más que un diagrama de clases extremadamente simplificado, en el cual se incluyen solamente los objetos que pertenecen a la vida real cuyo comportamiento o datos son relevantes en el sistema.

A partir de este pequeño modelo, se realiza un documento con los prototipos basándose en la interfaz gráfica, el cual se mostrará al cliente y se refinará en sucesivas reuniones. Normalmente este prototipo suele converger en dos o tres iteraciones.

Teniendo el documento de prototipos finales se y obtenidos todos los requisitos del usuario, se procede a la realización de los casos de uso. Estos diagramas se crean para obtener la trazabilidad.

4.5.3.2 Análisis y diseño preliminar

En esta segunda etapa ya se tienen los prototipos finales y los requisitos del sistema por parte del cliente, por tanto se procede a realizar los casos de uso. Después será necesario realizar lo que se conoce como Diagrama de Robustez, el cual pertenece al proceso ICONIX y tampoco forma parte del UML.

4.5.3.3 Diseño

En esta fase se proceden a realizar los diagramas de secuencia, los cuales derivan directamente de directamente de los caso de uso. Obsérvese como, los diagramas de secuencia se relacionan con los casos de uso que se relacionan con requisitos.

En caso de que el resultado no sea el esperado, será necesario repasar todo el proceso hasta que éste sea correcto. Es vital que los requisitos se satisfagan correctamente para el éxito del proyecto.

En este proyecto se realizó un artefacto más en esta etapa, el diagrama de base de datos, siendo este imprescindible para el correcto análisis y diseño de la aplicación.

4.5.3.4 Implementación

En esta etapa puede ser adecuado realizar un diagrama de componentes en algunos casos, pero no siempre es necesario. En cualquier caso, aquí es donde se escribe el código tal y como fue especificado en las fases anteriores y se planean las pruebas.

5. Estado del arte

Debido a la necesidad latente de autoevaluación y mejora de los procesos en las instituciones educativas se ha optado por implementar un tipo de pruebas de carácter ágil, flexible, dinámico, controlable y que permiten obtener resultados y almacenarlos en un repositorio de datos en tiempo real, es decir, las pruebas virtuales.

Unas pruebas virtuales básicamente consisten en un aplicación de carácter web o de escritorio con cliente/servidor, que almacenan preguntas de diferentes áreas en un banco de preguntas con un conjunto de posibles respuestas y su solución real, y que permite a un usuario de la aplicación acceder a dichas pruebas, enviar sus respuestas y recibir un informe de los resultados obtenidos.

Como antecedentes de implementaciones de dichas pruebas se pueden nombrar algunas aplicaciones de internet, como la denominada “Pre-ICFES virtual para preparar tu futuro” la cual es una de las aplicaciones más conocidas de este tipo y que permite la realización de pruebas y la visualización los resultados obtenidos, pero tiene como principal desventaja que no muestra los datos históricos que permitan identificar los avances o retrocesos de un usuario en las pruebas.¹¹

En la Universidad del Valle Sede Tuluá se cuenta con un antecedente de pruebas virtuales consistente en un simulador de pruebas SABER-PRO realizado con el fin de brindar al estudiante universitario que se encuentra próximo a realizar las pruebas SABER-PRO una opción para prepararse y conocer más a fondo dichas pruebas. Entre las principales características con las que cuenta este simulador están: es una aplicación web, permite crear reportes que muestran el rendimiento general o por área de los estudiantes en las pruebas realizadas y permite visualizar el resultado de cada estudiante acompañado de una gráfica estadística donde se muestra el comportamiento que tuvo este tras una prueba.¹²

La Institución Educativa Primavera no utiliza en la actualidad ninguna de las aplicaciones mencionadas, la aplicación “Pre-ICFES virtual para preparar tu futuro” no genera los reportes estadísticos que ellos desean, además la base de datos de las preguntas no puede ser actualizada por ellos tienen que valerse de la base de datos de la aplicación; mientras que la aplicación SABER-PRO tiene un enfoque completamente contrario a las necesidades de la institución.

¹¹<http://www.preicfesvirtual.net/contenidos/plPublic.php>.

⁸ Cardona Álvarez, A. Salamanca Soto, L. Desarrollo de una herramienta WEB simuladora de pruebas SABER-PRO, incorporando datos estadísticos para la toma de decisiones. Tuluá: Universidad del Valle. Facultad de **Ingeniería de Sistemas**; 2012

La principal diferencia entre el simulador de pruebas SABER-PRO y el aplicativo que se desea desarrollar para las pruebas pre-ICFES se basa en que las calificaciones de las pruebas SABER-PRO se realizan de una manera distinta y los resultados se generan con un enfoque estadístico diferente.

La aplicación de pre-ICFES formará parte esencial dentro del sistema para la toma de decisiones a desarrollar, al aportar los datos sistematizados de los resultados de las diferentes pruebas que se realicen y que servirán para poblar el data mart o bodega de datos tras un proceso de datawarehousing, que finalmente desencadenará en una minería de datos con el objetivo de la búsqueda de información relevante para el apoyo a la toma de decisiones dentro de la Institución Educativa Primavera.

Existe un antecedente de diseño de data warehouse en la Universidad del valle sede Tuluá (denominado UNISAD), en el cual se decidió implementar un data mart para los cursos de extensión de la universidad, entre los posibles datamarts que se tenían diseñados, es decir, extensión, secretaría académica y biblioteca, debido a que era uno de los que más información y procesos tenía para la toma de decisiones.

Para el diseño del datawarehouse de UNISAD, se realizó primero el croquis la base de datos transaccional de la sede, lo que incluye dependencias como extensión, secretaría académica, coordinaciones de planes, biblioteca y bienestar Universitario, ya que no estaba bien estructurada. Como resultado se obtuvieron consultas que permitían conocer entre otros cuestionamientos, el impacto de un determinado tipo de actividad en un tipo de población específica y la demanda tanto de actividades como de egresados que presentan las diversas empresas de la región. Las respuestas a dichas consultas además de presentar los resultados, eran soportadas por diversos tipos de gráficos estadísticos.¹³

CAPITULO II

6. Análisis

6.1 Metodología

6.1.1 Escogencia del tipo de metodología (ágil o tradicional)

Ambas metodologías son importantes y de gran aceptación en el mundo, se decidió por la teoría ágil debido a que:

- Se necesitaba una metodología que diera la oportunidad cambios de forma sencilla durante el proceso, ya que los requisitos del usuario eran muy susceptibles a modificaciones.
- Se necesitaba un proceso muy ágil con pocas políticas y que no fuese tan riguroso en cuanto a la creación de artefactos.
- Ya que el proceso debía ser rápido era necesario que no hubiese una gran cantidad de artefactos, pero si los suficientes para hacer un correcto análisis y diseño del sistema.
- La metodología escogida debería tener muy pocos roles definidos esto debido a que el grupo de desarrollo era pequeño.

⁹ Cubillos Jurado, A. Escobar Tobar, J. Desarrollo de un data warehouse e implementación de un data mart para la Universidad del Valle sede Tuluá. Tuluá: Universidad del Valle. Facultad de Ingeniería de Sistemas; 2006.

- La metodología debería estar enfocada a grupos de desarrollos pequeños y que trabajen en un mismo sitio.

6.1.2 Escogencia entre las tres grandes metodologías ágiles

ICONIX es la metodología escogida, a pesar de que XP y SCRUM cumplen con los requisitos que se necesitan para este desarrollo, ICONIX ofrece una documentación un poco más detallada en el proceso de diseño, además de utilizar artefactos de UML un lenguaje en el cual el equipo desarrollador tiene cierta experiencia.

En la etapa de análisis ICONIX define los siguientes artefactos:

- Modelo de dominio
- Prototipado
- Diagrama de paquetes

El modelo de caso de uso se realizó en la próxima etapa en paralelo con la descripción de los mismos.

6.2 Modelo de dominio

6.2.1 ¿Que es el modelo de dominio?

El modelo de dominio es un artefacto perteneciente al análisis, se construye con las reglas de UML durante la fase de inicio, en la tarea de construcción del modelo, se presenta como un diagrama de clases que contiene no conceptos propios de software si no de la propia realidad física.

Los modelos de dominio se utilizan para capturar y expresar el entendimiento sobre el proceso que se ha ganado en la etapa de análisis de los requisitos dados por el usuario que siempre es la etapa anterior a plantear cualquier artefacto del proceso, el modelo de dominio es utilizado para entender el sector en el cual se encuentra el sistema a realizar.

El paso esencial en el enfoque orientado a objetos es descomponer el problema en objetos reales e individuales, el modelo representa los conceptos del dominio del sistema; en el modelo de dominio no se puede especificar ninguna operación del sistema.

Pero ¿qué es un concepto? Un concepto en términos generales es una idea, cosa u objeto. Aunque definido más formalmente es una unión entre un símbolo, una intención y una extensión.

- Símbolo: Palabras o imágenes que representen un concepto.
- Intensión: Se define el concepto.
- Extensión: El conjunto de ideas que se aplica al concepto.

Un modelo conceptual muestra:

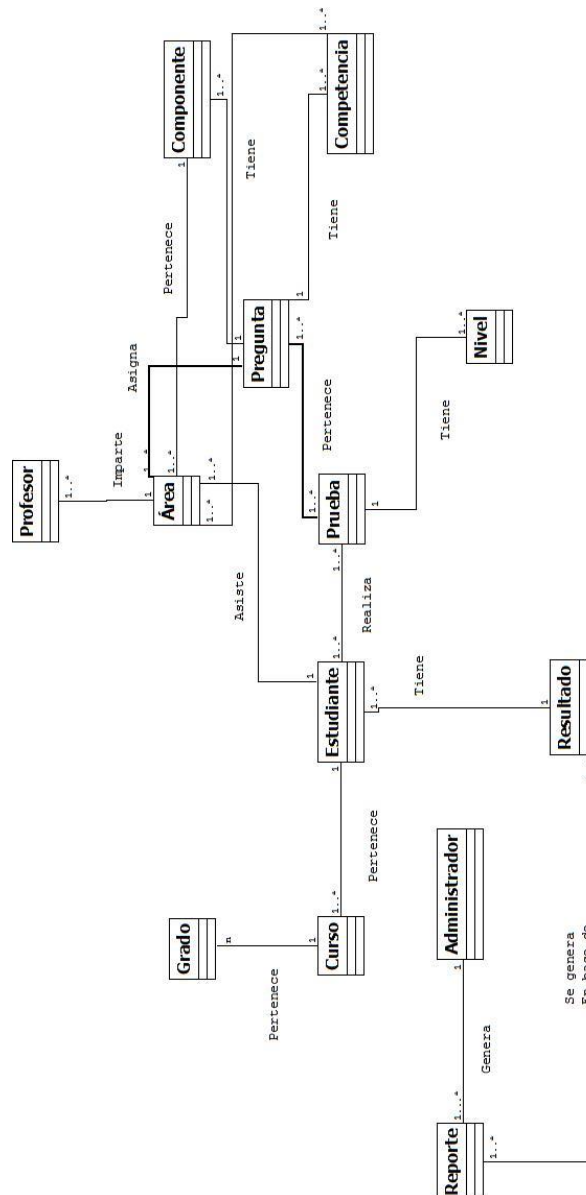
- Conceptos generales del sistema.
- Asociaciones entre conceptos.
- Atributos de conceptos.

6.2.2 ¿Qué representa y que no el modelo de dominio?

El modelo de dominio permite descomponer el campo del problema en conceptos reales y comprensibles, lo que permite definir más claramente la terminología y la nomenclatura del sistema.

El modelo de dominio no es una descripción del diseño del software, como una clase en C++, una ventana, una base de datos, etc.

6.2.3 Modelo de dominio del sistema.



(Figura 1)

6.2.4 Explicación modelo de dominio

- El administrador puede generar muchos reportes, además el reporte es generado por un solo administrador ya que en el sistema solo existe uno.
- Un estudiante pertenece a un solo curso, mientras que un curso le pertenecen muchos estudiantes, además de esto cada curso es asignado a un grado y evidentemente a un grado se le asignan muchos cursos.
- Un estudiante tiene muchos resultados ya que presenta muchas pruebas durante su paso por la institución, pero cada resultado pertenece a un solo estudiante. Además vale la pena aclarar que los resultados serán almacenados por separado para cada componente, competencia y área.
- El estudiante asiste a muchas áreas y a cada área tiene muchos estudiantes, además cabe aclarar que cada área es impartida por un solo profesor pero un profesor puede impartir muchas áreas.
- Una misma prueba es presentada por múltiples estudiantes, a cada prueba se le asigna uno y un solo nivel, además de muchas preguntas.
- Cada pregunta se le asigna un componente y una competencia, además de un área, y claramente tanto el componente, como competencia, como el área pueden ser asignados a muchas preguntas.
- Un reporte se genera en base a muchos resultados, y un resultado es base de muchos reportes,

6.2.5 Importancia del modelo de dominio en el proceso

El modelo de dominio fue de vital importancia en el proceso, ya que en este se comprendieron claramente los conceptos del sistema, además de ser un insumo vital para la creación de el diagrama de clases y el diagrama de base de datos.

6.3 Modelo de prototipos

El modelo de prototipos los cuales se utilizan para dar al usuario una vista preliminar de parte del software. Este modelo es básicamente prueba y error ya que si al usuario no le gusta una parte de la interfaz significa que la prueba fallo por lo cual se debe corregir el error que se tenga hasta que el usuario quede satisfecho.

Existen dos clases de prototipos el desechable y el evolucionario; el desechable sirve para eliminar las dudas sobre lo que quiere el cliente además se utiliza para desarrollar la interfaz que más convenga al cliente, mientras el evolucionario es un modelo que pasa de ser un prototipo a ser software pero no tiene ni una buena documentación ni una buena calidad.

El documento de prototipos utilizado por la metodología ICONIX y por ende el utilizado en nuestro modelo solo maneja prototipos desechables.

La realización del modelo de prototipado por lo general se realiza en las siguientes etapas:

- Recolección y refinamiento de requisitos
- Modelado
- Construcción del Prototipo
- Desarrollo, evaluación del prototipo por el cliente
- Refinamiento del prototipo

Recolección y refinamiento de requisitos: En esta etapa se estudian los requisitos del cliente, se analizan y se mejoran para que expresen claramente lo que el cliente necesita.

Modelado: En esta etapa se diseña en papel y con la ayuda del cliente la interfaz que va a representar cada prototipo.

Construcción del prototipo: Con ayuda de una herramienta de procesamiento de imágenes se construye de la forma más ágil y sencilla posible la interfaz para mostrar al usuario.

Desarrollo: El prototipo construido es mostrado al cliente para que el decida si se ajusta a sus necesidades.

Refinamiento del prototipo: Si la prueba con el cliente no fue exitosa se corrigen los errores y se regresa a la etapa anterior.

6.3.1 Prototipos del sistema

6.3.1.1 Prototipo de autenticación

Prototipo 1	
Historia de usuario N°: 1	Nombre: Autenticación
Resumen: Solicitud para ingresar a la aplicación.	
 Iniciar sesión   	

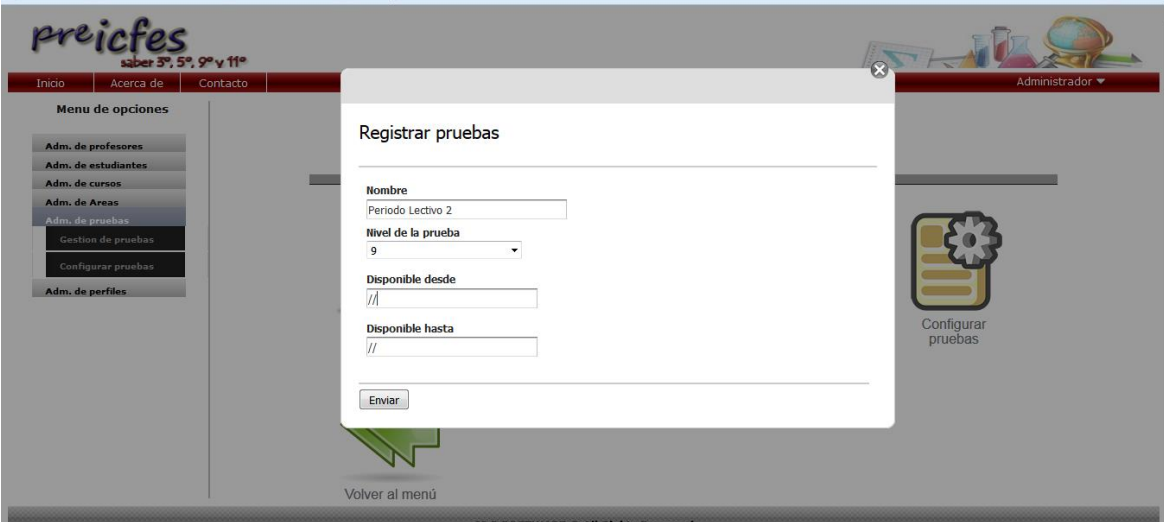
(Figura 2)

Una breve explicación: En la primera parte se encuentra el número del prototipo que indica el orden en el que fueron creados, en la segunda la historia de usuario que se cumple con este prototipo, después se tiene el nombre y un resumen de lo que se hace en la interfaz. Finalmente se ubica el prototipo o interfaz.

En definitiva: Este diagrama de prototipos muestra la autenticación del usuario, aquí se encuentra los tres tipos de usuarios que existen en el sistema, además del logo de la aplicación. La prueba con el usuario fue exitosa y este prototipo fue el definitivo y esta interfaz fue construida dentro del sistema.

6.3.1.2 Prototipo de creación de prueba

Prototipo 15	
Historia de usuario N°: 6.1	Nombre: Crear Prueba
Resumen: Se crea una prueba para que se le puedan incluir preguntas por parte de los maestros.	



The screenshot displays the 'preices' web application interface. A modal window titled 'Registrar pruebas' is centered, containing a form with the following fields: 'Nombre' (text input with 'Periodo Lectivo 2'), 'Nivel de la prueba' (dropdown menu with '9'), 'Disponible desde' (text input with '//'), and 'Disponible hasta' (text input with '//'). An 'Enviar' button is located at the bottom of the form. The background shows the application's main interface, including a sidebar menu with options like 'Adm. de profesores', 'Adm. de estudiantes', 'Adm. de cursos', 'Adm. de Areas', 'Adm. de pruebas', 'Gestion de pruebas', 'Configurar pruebas', and 'Adm. de perfiles'. A 'Configurar pruebas' button is also visible on the right side of the main content area.

(Figura 3)

En definitiva: Este prototipo muestra la creación de una prueba por parte del administrador, aquí se encuentra el formulario de creación resaltado en la interfaz además de la página principal de administrador en la parte de atrás. Este prototipo en comienzo sufrió algunos cambios ya que en principio no se consideró la disponibilidad de la prueba.

6.3.3 ¿Por qué es importante el prototipado dentro del sistema?

El prototipado fue importante en el sistema ya que gracias a él se pudo garantizar que todas las interfaces cumplieran con las especificaciones y las necesidades del usuario.

6.4 Diagrama de paquetes

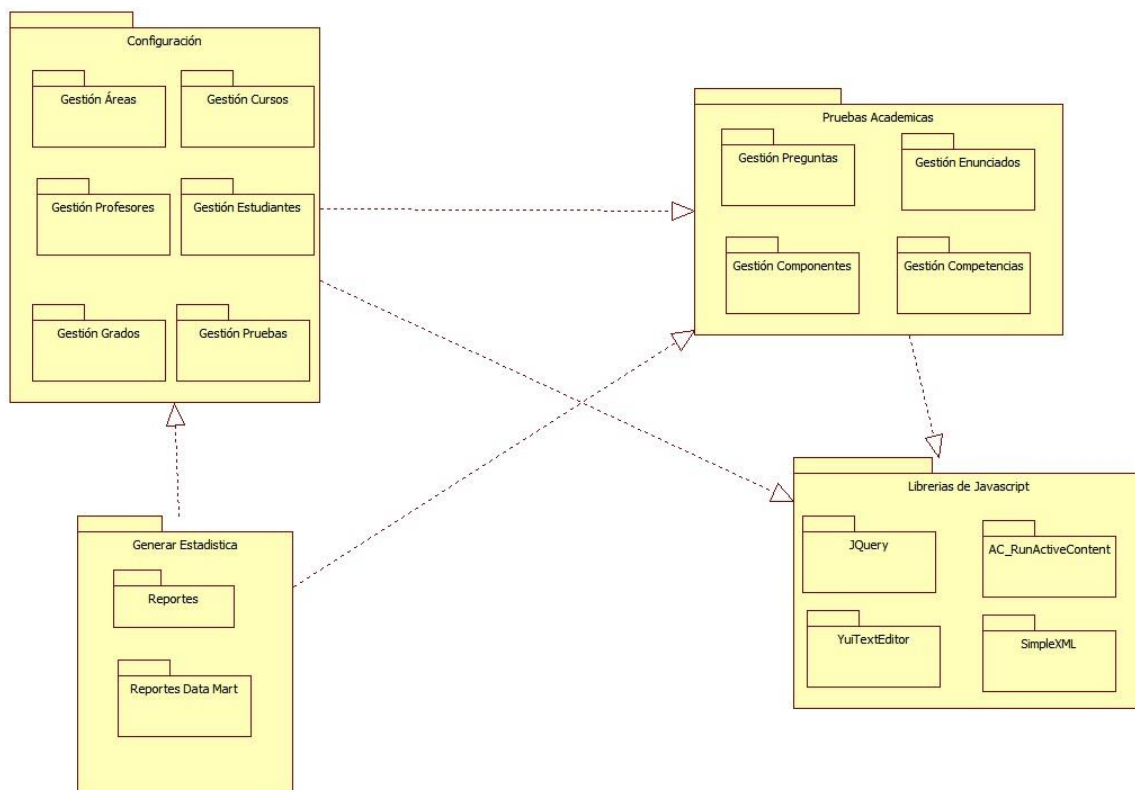
Un diagrama de paquetes muestra la forma en que el sistema está dividido en agrupaciones lógicas mostrando las dependencias entre estas agrupaciones. Igualmente suministra la descomposición de la jerarquía y lógica de un sistema.

Los paquetes están organizados para maximizar la coherencia de cada paquete y minimizar el acoplamiento externo entre los paquetes. Estos son elementos de gestión que ayudan a comprender el orden en el que se debe desarrollar cada elemento.

Generalidades de los diagramas de paquetes:

- Permite generar la arquitectura del sistema de forma macro, muestra la estructura del sistema.
- Además cada paquete puede contener sub paquetes que tienen interfaces y realizan alguna funcionalidad.
- Es importante evitar dependencias circulares dentro de los paquetes y esto siempre es posible en cualquier sistema, a diferencia de las clases que en ocasiones no se puede evitar.

6.4.1 Diagrama de paquetes del sistema (Figura 4)



(Figura 4)

6.4.2 Importancia del diagrama de paquetes dentro del sistema

Por medio de este diagrama se logró conocer de forma general la estructura del sistema, además de permitir especificar el orden en que debió irse creando el sistema, y definir ya de forma más clara los módulos a realizar.

7. Análisis y diseño preliminar

Para el diseño ICONIX define dos artefactos:

- Descripción de los casos de uso.
- Diagramas de robustez.

Como se mencionó anteriormente se realizó el modelo de casos de uso en esta etapa.

7.1 Casos de uso

7.1.1 Introducción

Cada caso de uso constituye el flujo completo de eventos en el sistema, que especifican la interacción que toma lugar entre el actor y el sistema.

Un diagrama de casos de uso muestra, los distintos requisitos funcionales (solo los requisitos funcionales) que se esperan del sistema y cómo se relacionan con su entorno (el entorno puede ser usuarios u otras aplicaciones).

Definiciones importantes para el modelo de caso de uso:

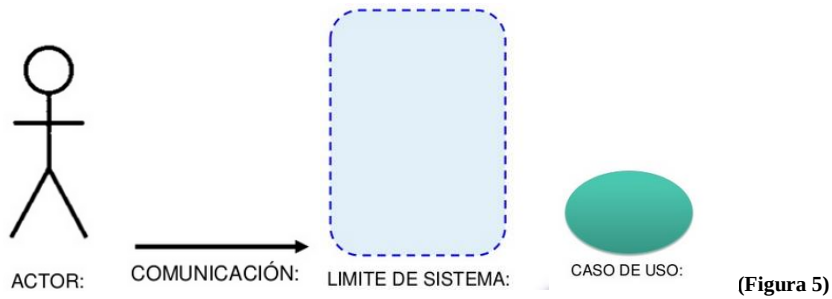
Actor: Representa quien o que inicia una acción dentro del sistema, es el rol que lleva una persona o cosa. Un actor en el diagrama se representa con la figura de una persona.

Comunicación: Este elemento representa la relación que existe entre el caso de uso y un actor, dicho elemento se representa simplemente por una línea recta que se extiende desde el caso de uso hasta el actor.

Límites del sistema: Empleado para delimitar los límites del sistema, y representado por un rectángulo de color con un fondo distinguible.

Caso de uso: Representa a quien o que inicia una acción dentro del caso de uso, es una tarea que debe llevarse a cabo en el sistema que se está detallando, se representa mediante un ovalo con un color que se distinga del fondo.

La figura 5 muestra los artefactos anteriormente nombrados.



7.1.2 Descripción de un caso de uso del sistema

Caso de uso: Editar la configuración de una pregunta.

Actores: Profesor.

Propósito: Permitir al profesor editar la configuración inicial de una pregunta.

Resumen: El profesor desea modificar la configuración inicial de una pregunta su título, nivel de la prueba área evaluativa, competencia, componente o grado.

Curso normal de los eventos	
Usuario	Sistema
1. Selecciona las opciones de pregunta.	2. Despliega las opciones de pregunta
3. Indica el deseo de actualizar una pregunta	4. Despliega la lista de todas las preguntas.
5. Indica la pregunta a la cual desea cambiar su configuración.	6. Despliega el formulario de configuración de una pregunta.
7. Modifica el título, nivel de la prueba área evaluativa, competencia, componente o grado.	9. Valida la información ingresada.
8. Confirma que desea modificar la pregunta elegida.	10. El sistema devuelve al usuario a la página principal.

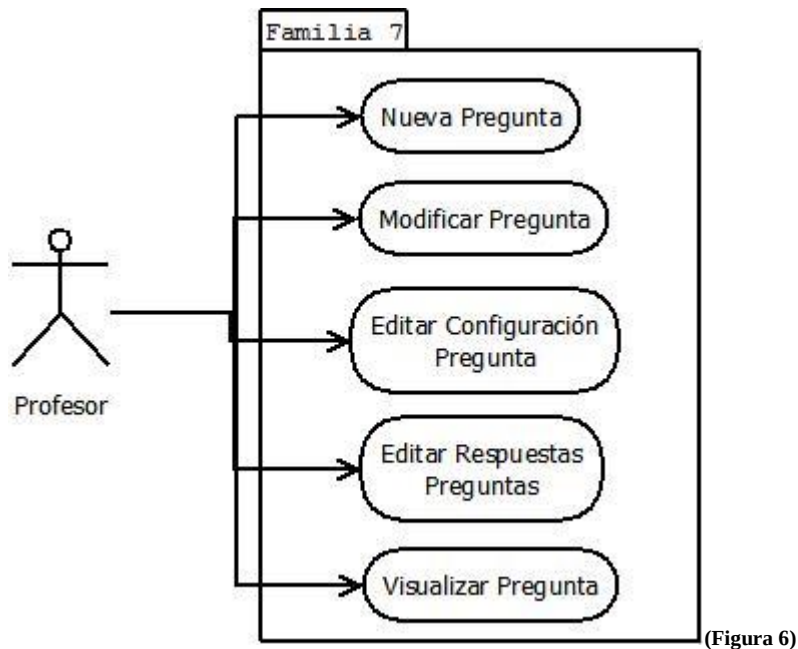
Curso alterno y Excepciones
9.1 La información ingresada no es válida (véase, tabla3), por lo que el sistema solicitará su modificación.

(Tabla 4)

7.1.2.1 Importancia de la descripción de casos de uso

Esta descripción muestra cómo debe desarrollarse el sistema, el código se realiza para que cumpla con estas especificaciones. Además de ser un apoyo indispensable para las pruebas ya que el cliente deberá de ejecutar estos casos y comprobar que la aplicación se comporta como allí se describe.

7.1.3 Ejemplo modelo de caso de uso del sistema (Figura 6).



(Figura 6)

7.1.3.1 Importancia de los modelos de casos de uso

Este modelo muestra cómo interactúan cada uno de los usuarios con la aplicación, lo que le proporciona al desarrollador una base para realizar la codificación.

7.2 Modelo de robustez

Un diagrama de robustez es sencillamente un híbrido entre el diagrama de clases y un diagrama de actividades.

El diagrama de robustez es una herramienta que permite capturar el qué hacer y después ayuda a identificar cómo hacerlo.

7.2.1 Una breve introducción al diagrama de robustez

Un diagrama de robustez se compone de tres objetos que son:

- **Objetos Fronterizos:** Estos objetos indican los cambios de interfaz para el usuario.
- **Objetos Entidad:** Indican interfaces o entidades del modelo de dominio.
- **Objetos Control:** Permiten la unión de los demás objetos dentro del sistema (En la mayoría de los casos son botones).

En la figura 7 se muestra los artefactos que se incluyen en el diagrama de robustez.

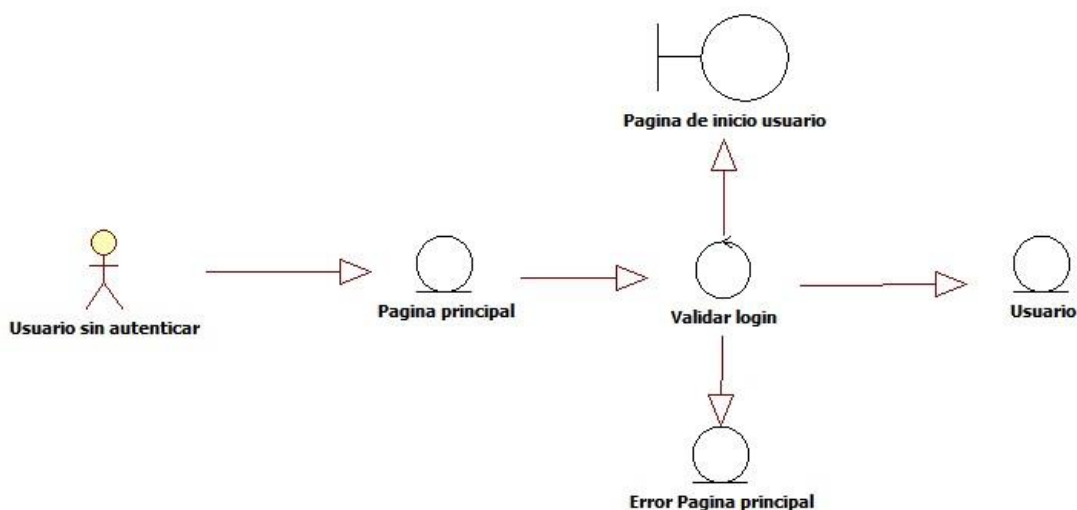


(Figura 7)

7.2.2 Diagrama de robustez de autenticación (Figura 8)

El siguiente diagrama cumple con los casos de uso de:

- Autenticar usuario.



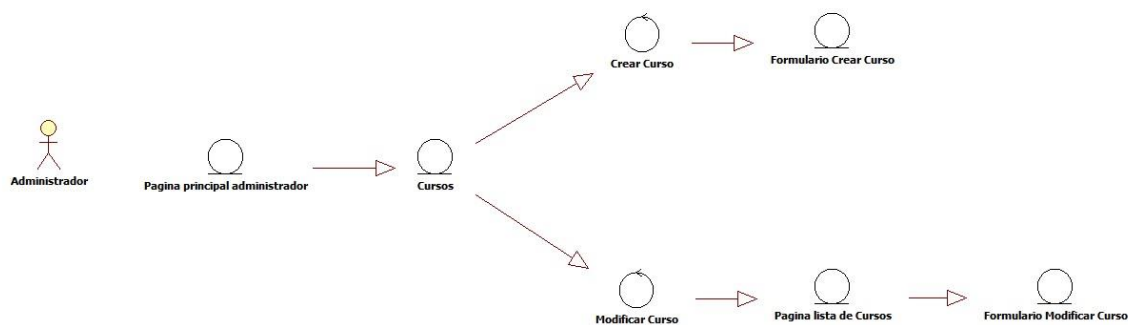
(Figura 8)

Breve explicación: Un usuario del sistema ingresa a la página principal, en esta ingresa su información personal (administrador o el profesor ingresa su contraseña y si es estudiante ingresa su login y su contraseña), el sistema valida la información ingresada, si la información es correcta el usuario sin autenticar pasa a ser un usuario; si no es correcto lo devuelve a la página principal.

7.2.3 Diagrama de robustez de cursos (Figura 9)

El siguiente diagrama cumple con los casos de uso de:

- Crear curso
- Modificar curso



(Figura 9)

Breve explicación: El administrador que es el único usuario que tiene la posibilidad de manejar los cursos se encuentra en su página principal, de allí accede a la página de cursos y en esta decide si crea o modifica un curso, si decide crearlo se genera el formulario de curso, si por el contrario decide modificar el curso se genera la lista de cursos y allí escoge el curso a modificar.

7.2.4 Importancia de los diagramas de robustez en el sistema

Los diagramas de robustez ayudan a comprender que se debe hacer al mostrar claramente las interfaces y los objetos que se deben construir, además muestra la unión de estas interfaces y objetos lo que indica el cómo se debe realizar la aplicación.

8. Diseño

En la etapa de diseño ICONIX define los siguientes artefactos:

- Diagrama de secuencia
- Modelo de base de datos
- Diagrama de clases

Se agrega de igual manera toda la parte del diseño del datamart que será el insumo vital de la minería de datos.

8.1 Diagrama de secuencia

Los diagramas de secuencia se definen para describir gráficamente un caso de uso o un escenario; además de definir los módulos o clases que van a formar parte del programa y las llamadas que se hacen para realizar una tarea determinada.

8.1.1 Una breve explicación del diagrama de secuencia

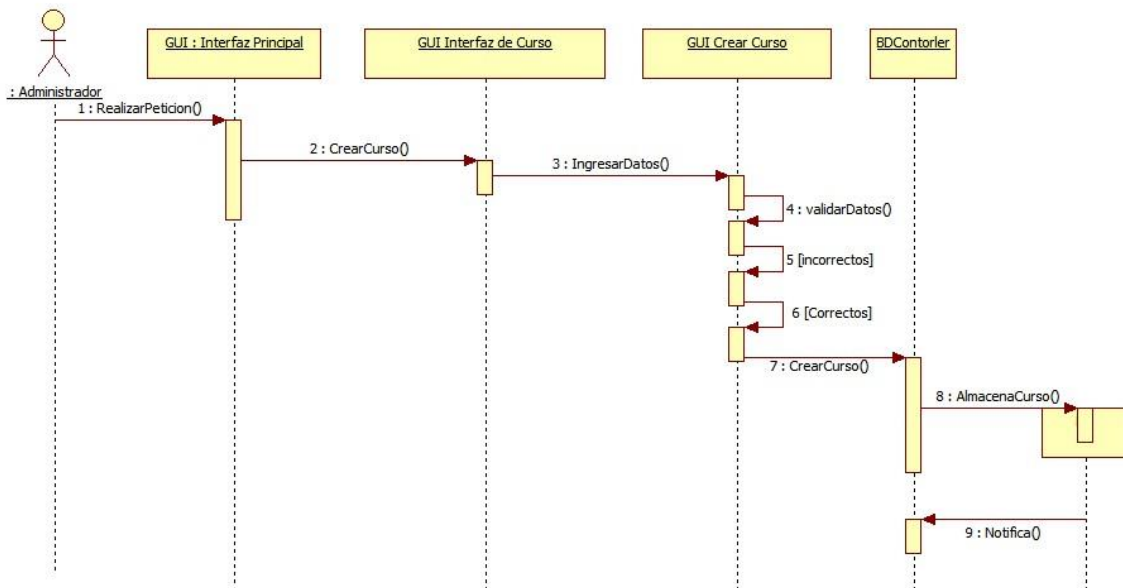
El diagrama de secuencia se contiene cuatro objetos importantes que son:

- **Clase:** Este objeto contiene el nombre de la clase, objeto o base de datos que intervienen en el sistema.
- **Mensaje:** Define el mensaje u operación que se realiza entre las diferentes clases del sistema, además señala la dirección que lleva la aplicación en cada momento.

- **Barra de sincronización temporal:** Indica el tiempo de vida del objeto dentro del sistema, aunque por lo general se dibujan hasta el final de la aplicación (lo que no siempre es correcto), además de indicar la vida del mensaje en la aplicación.
- **Actor:** Es el encargado de comenzar todo el proceso que el diagrama quiere ilustrar.

8.1.2 Diagrama de secuencia de crear un curso

Representa el caso de uso de crear curso.

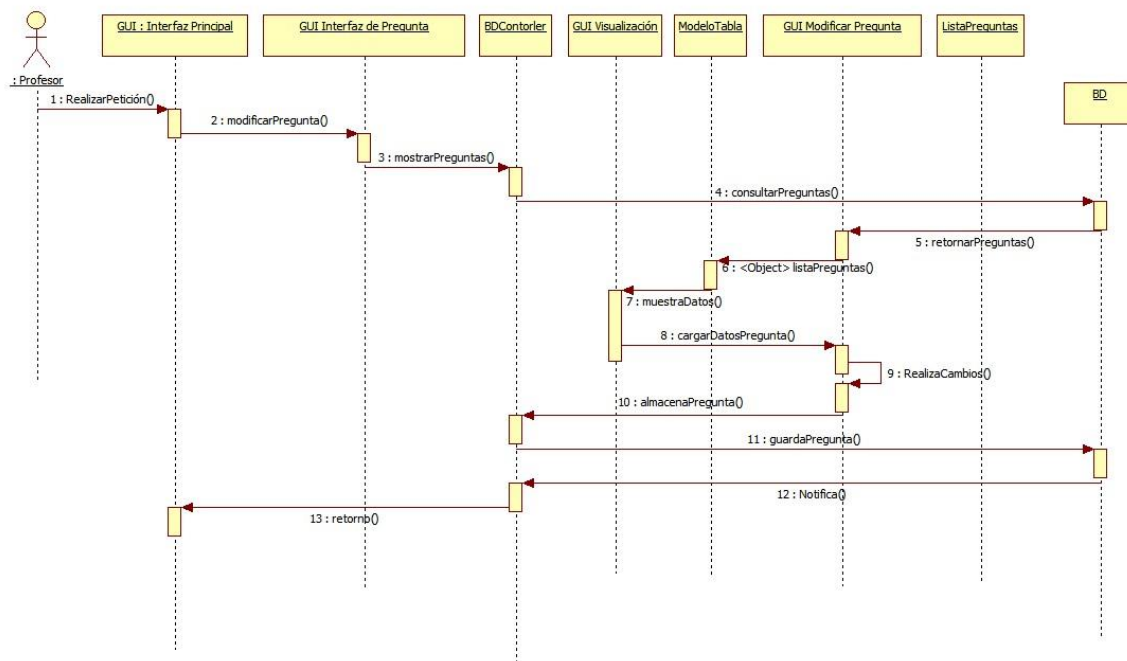


(Figura 10)

Breve explicación: El actor es el administrador el único que tiene la capacidad de crear curso, este accede a la interfaz principal que corresponde a su perfil allí crea el curso en la interfaz de curso, ingresa los datos el sistema valida estos datos, si no son correctos devuelve al usuario a la página de crear curso, si son correctas se almacena y se notifica al usuario.

8.1.3 Diagrama de secuencia de modificar pregunta

Representa el caso de uso de modificar pregunta.



(Figura 11)

Breve explicación: El recorrido del sistema es igual al caso anterior, pero se incluye un objeto diferente que es la lista de preguntas, este objeto se alimenta de la base de datos y es mostrado al usuario de forma organizada por la clase ModeloTabla.

8.1.3 Importancia de los diagramas de secuencia en el proceso

Los diagramas de secuencia fueron de gran importancia en el proceso ya que muestran el orden que se debe llevar en cada uno de los procesos que se realizan en el sistema.

8.2 Esquema de base de datos

Es un modelo en donde se plasma las tablas que posteriormente serán creadas en la base de datos además de las relaciones entre ellas, y la cardinalidad de cada una de estas relaciones.

8.2.1 Esquema de base de datos del sistema

8.2.2 Importancia del esquema de base de datos en el sistema

El modelo de la base proporciona la base para la creación de la base de datos final del sistema, además de ilustrar claramente la relación de los objetos virtuales que interactúan en el sistema.

9. Implementación

ICONIX en la etapa de implementación define los siguientes artefactos:

- Diagrama de componentes
- Documento de pruebas

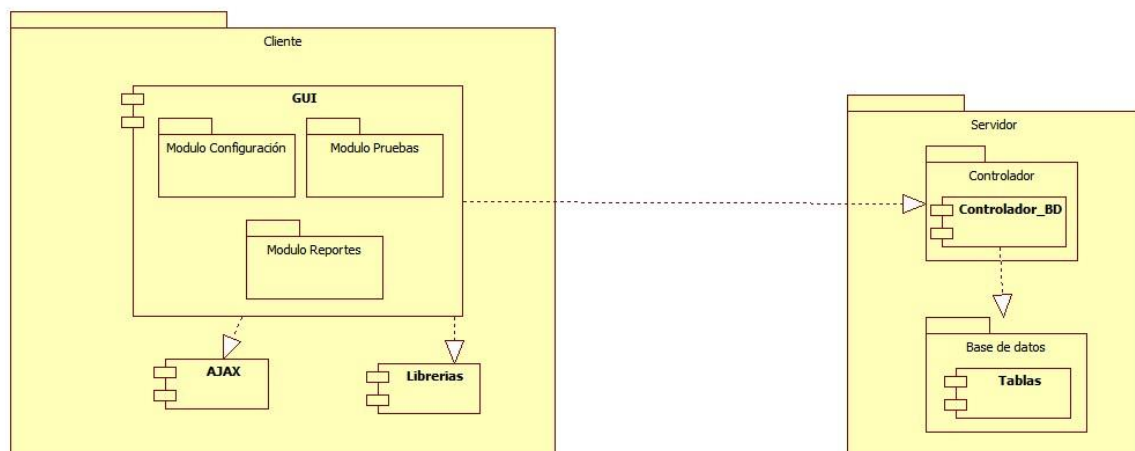
9.1 Diagrama de componentes

El diagrama de componentes define los elementos físicos del sistema y la relación que existe entre estos.

Hay cinco estereotipos estándar que se aplican a los componentes y estos son:

- **Ejecutable:** Especifica un componente que se puede ejecutar en un nodo.
- **Librería:** Especifica una biblioteca de objetos bien sea estática o dinámica.
- **Tablas:** Especifica un componente que representa una tabla de la base de datos.
- **Archivo:** Especifica un componente que representa un documento que contiene código fuente o datos.
- **Documento:** Especifica un componente que representa un documento.

9.1.1 Diagrama de componentes del sistema



(Figura 15)

Breve explicación: Se tienen dos nodos el nodo cliente y el nodo servidor, en el nodo cliente se encuentra con el componente GUI, allí se encuentran los grandes módulos de la aplicación, además de esto se tiene dos componentes adicionales que son la librería de Ajax, uno donde se encuentra todas las librerías utilizadas. En el nodo servidor se tiene el controlador de la base de datos y la base de datos como tal.

9.2 Documento de pruebas

El objetivo de este documento es el de confeccionar y ejecutar un plan de pruebas para descubrir el mayor número de errores posibles.

Se siguen los casos de uso ya especificados. De esta forma se comprueba que, suponiendo unos datos correctos, los resultados obtenidos son los esperados.

9.2.1 Prueba Crear Profesor

Se satisface el caso de uso de crear profesor

Nombre	Crear un profesor		
Propósito	Comprobar si se realiza correctamente el registro de un profesor		
Prerrequisito	El profesor no debe de estar registrado		
Pasos	1. Página principal de administrador 2. Click en profesor 3. Click en nuevo 4. Diligenciar formulario 5. Click en guardar 6. Verificar mensaje de registro exitoso		
Ejemplo			
Datos de entrada		Datos de salida	Resultado esperado
Nombre: Leidy Mariana Apellido: Castañeda Ortiz Correo electrónico:lindabb1@gmail.com Dirección: Calle 18 Número 10-46 Fecha de nacimiento: 26/10/1993 Teléfono: 3154587896 Código interno: 124568			Exitoso

(Tabla 7)

CAPITULO III

10. PRUEBAS ACADEMICAS

10.1 Banco de datos

La aplicación permite al usuario almacenar las preguntas en un banco de datos para utilizarlas posteriormente en la realización de las pruebas.

El ICFES maneja dos tipos diferentes de preguntas en la realización de sus pruebas:

- Selección múltiple con única respuesta (Tipo I)
- Selección múltiple con múltiple respuestas (Tipo IV)

Además de esto el ICFES utiliza enunciados a los cuales se les asigna preguntas; cabe anotar que los enunciados, preguntas y respuestas pueden ser textos o imágenes.

10.1.1 Preguntas tipo I

En este tipo de pregunta se crea un encabezado y se dan las cuatro (4) opciones de respuesta, las cuales tienen un porcentaje de correctitud y el estudiante puede escoger una y sola una opción de respuesta, un ejemplo claro de este tipo de pregunta sería:

- *“Cuando un hombre pretende ser no violento no debe irritarse contra aquel que lo ha ultrajado. No le deseará ningún mal; le deseará bien; no lo maldecirá y no le infligirá ningún sufrimiento físico. Aceptará todos los ultrajes que le haga sufrir el ofensor. La no-violencia, comprendida de esta forma, se convierte en la inocencia absoluta. (Mahatma Ghandi).El concepto de la no-violencia que defendió Ghandi, implica:*
 - A. *Humildad y sencillez de corazón. **Correctitud = 100%***
 - B. *Que cualquier tipo de opresión es innecesaria. **Correctitud = 25%***
 - C. *Las guerras de liberación son baños de sangre. **Correctitud = 0%***
 - D. *Que la acción política no es igual a pasividad o ingenuidad. **Correctitud =0%**”¹⁴*

La aplicación permite la posibilidad de crear este tipo de preguntas, ingresando originalmente desde el rol de profesor, allí se genera un editor que permite el ingreso del encabezado de la pregunta, además el sistema solicita la información de configuración de la pregunta y las respuestas con su respectiva correctitud.

[1] ICFES , Ejemplo de preguntas, se encuentra en http://www.icfes.gov.co/examenes/index.php?option=com_content&view=article&id=156&catid=21&Itemid=206

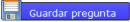
1. En la imagen uno (1) se muestra el editor de texto en el que se ingresa el encabezado de la pregunta.

Nueva pregunta

Font Name and Size: Arial 18. Font Style: Bold, Italic, Underline, Text Color, Background Color, Bulleted List, Numbered List. Undo/Redo, Alignment: Left, Center, Right, Justify. Paragraph Style: Normal. Indenting and Lists: First Line Indent, Hanging Indent, Bulleted List, Numbered List. Insert Item: Image, Table, Link, Unlink.

Cuando un hombre pretende ser no violento no debe irritarse contra aquel que lo ha ultrajado. No le deseará ningún mal; le deseará bien; no lo maldecirá y no le infligirá ningún sufrimiento físico. Aceptará todos los ultrajes que le haga sufrir el ofensor. La no-violencia, comprendida de esta forma, se convierte en la inocencia absoluta. (Mahatma Ghandi).El concepto de la no-violencia que defendió Ghandi, implica:

body < p.MacList&P... < span < span



2. En la imagen dos (2) se observa la configuración de una pregunta recién creada.

Configurar pregunta [X]

Configurar pregunta ▾

Título
Ghandi

Nivel de la prueba
11

Componente evaluativo
Núcleo común

Area evaluativa
Ciencias Naturales

Competencia
Lectora

Componente
[]

Grado
11

Asignar respuestas ▾



3. En la imagen tres (3) se muestra la configuración de las respuestas asignadas a la pregunta recién creada.

Configurar pregunta [X]

Configurar pregunta ▾

Asignar respuestas >

Respuesta	Correctitud (%)
Respuesta 1 Humildad y sencillez de corazón. 	100
Respuesta 2 Cualquier tipo de opresión es innecesaria. 	25
Respuesta 3 Las guerras de liberación son baños de sangre. 	0
Respuesta 4 La no violencia no es igual a pasividad o ingenuidad. 	0



10.1.2 Preguntas tipo IV

En este tipo de pregunta se dan cuatro (4) respuestas de las cuales el estudiante debe escoger dos respuestas, y se escogen de la siguiente forma.

- Si 1 y 2 son correctas, rellene el óvalo A.
- Si 2 y 3 son correctas, rellene el óvalo B.
- Si 3 y 4 son correctas, rellene el óvalo C.
- Si 2 y 4 son correctas, rellene el óvalo D.

Un claro ejemplo de esta es:

- *“Para que un grupo de estudiantes de un colegio elabore un estudio ambiental, con el fin de mejorar las condiciones del establecimiento y su entorno, es recomendable:*
 1. *Promover actividades científicas y eventos culturales con otras ciudades.*
 2. *Recuperar las áreas recreativas y la infraestructura del establecimiento.*
 3. *Apropiarse de los espacios públicos dentro y fuera del establecimiento.*
 4. *Elaborar estudios técnicos sobre la valorización del suelo en la zona.”*¹⁵

Para realizar este tipo de preguntas el usuario la configura exactamente igual que el anterior, caso mostrado.

10.1.3 Enunciados

Los enunciados son en la mayoría de ocasiones lecturas, sobre las cuales se realiza preguntas de competencia lectora, aunque en algunas ocasiones dichos enunciados son imágenes o lecturas informativas para realizar preguntas de pensamiento e interpretación de imágenes.

En este pre-icfes virtual se almacenan estos enunciados y se da la posibilidad de asignarle preguntas, además de permitir al usuario en el momento de configurar las pruebas asignar las preguntas que deseen y no las que el enunciado tenga asignadas en su totalidad.

[1] ICFES , Ejemplo de preguntas, se encuentra en http://www.icfes.gov.co/examen/index.php?option=com_content&view=article&id=156&catid=21&Itemid=206

1. En la imagen cuatro (4) se muestra el editor de enunciados, en el cual se ingresa el cuerpo del enunciado

Nuevo enunciado

Font Name and Size: Arial, 13

Font Style: Bold, Italic, Underline

Text Color, Background Color, Undo/Redo, Alignment, Paragraph Style, Indenting and Lists, Insert Item

En un supermercado realizan una promoción que consiste en que por hacer una compra mayor de \$70.000, al comprador se le permite participar en un sorteo por una sola vez. El comprador que desee participar debe presentar su ticket de compra con el que podrá extraer de una bolsa una balota y de acuerdo con su color obtendrá un premio. El supermercado ha establecido algunas horas durante el día para realizar esta promoción y, de acuerdo con la hora, se jugará con una bolsa distinta, así:

BOLSA 1: 8: am a 10:00 am
BOLSA 2: 12:00 m a 2:00 pm
BOLSA 3: 5:00 pm a 7:00 pm

Legend:
○ No gana premio
■ Gana un mercado por un valor de \$100.000
▲ Gana un descuento del 20% en cualquier compra
◆ Gana un bono por \$10.000

body

Guardar enunciado

2. En la imagen cinco (5) se muestra la configuración de un enunciado nuevo.

Configurar enunciado

Titulo: Supermercado

Encabezado: Lea para que responda adecuadamente

Nivel de la prueba: 11

Componente evaluativo: Núcleo común

Área evaluativa: Ciencias Naturales

Grado: 11

✓ Agregar

10.2 Configuración de las pruebas.

Las pruebas son altamente configurables, dándole la posibilidad a la institución de configurar las pruebas según sus necesidades y procesos tanto los periodos lectivos, componentes, competencias, áreas, las áreas que se encuentran en cada componente evaluativo o la cantidad de preguntas máximas que se pueden evaluar de un área en una misma prueba.

La institución configura los periodos lectivos que internamente manejan, para posteriormente asignar profesores en estos periodos a cada área, además de que al crear la prueba deben de indicar a que periodo lectivo pertenece

En esta aplicación el usuario tiene la capacidad de crear competencias, componentes además de asignar estos a las áreas que están almacenadas en el sistema, para su posterior ingreso a las pruebas que se realizan finalmente a los estudiantes esto debe.

Las áreas pueden ser creadas, modificadas, agregadas a las pruebas, asignarle los componentes evaluativos a los que corresponde (los componentes evaluativos son: Núcleo común, Interdisciplinar y Profundización) y la cantidad de preguntas por prueba que se pueden agregar de dicha área a una prueba.

10.3 Reportes especiales (minería de datos)

La minería de datos en esta aplicación tiene el objetivo de proporcionar al usuario bases para la toma de decisiones institucionales, tales como:

- Enfocar el área de matemáticas para que los estudiantes fortalezcan el componente Numérico-Varacional en vez de que la indagación general de los estudiantes de igual manera mejore.
- En todas las materias que contengan la competencia argumentativa darle especial énfasis a esta, con la intención de mejorar en el resultado final de español.

10.3.1 Diseño del datamart

Inicialmente el objetivo número tres (3), indicaba la utilización de las calificaciones que el estudiante ha obtenido en un periodo para sobre estos datos realizar el proceso de minería, pero un análisis posterior conllevó a la conclusión que el cumplimiento de esta parte del objetivo implicaba la creación de una aplicación que se encargará de todo el proceso de carga y clasificación de notas estudiantiles, lo que implica un trabajo extra y que puede alargar el tiempo de implementación demasiado para el cronograma propuesto, además el beneficio que trae para el cumplimiento del objetivo es algo extra y obvio por el momento.

Si bien es cierto esta información puede ser de gran ayuda para otro tipo de análisis por lo que se plantea como trabajo futuro.

Nombre de la dimensión	Descripción de la dimensión
Estudiante	El estudiante es una parte primordial de la institución, además se encuentran directamente asociados a las calificaciones, lo que los convierte en materia prima primordial para nuestro datamart.
Componente	Los componentes son importantes ya que sobre ellos se sacaran las conclusiones.
Competencia	Al igual que los componentes, las competencias serán la base de las conclusiones obtenidas.
Áreas	Las áreas son fundamentales en nuestro modelo ya que ahí están asociadas las calificaciones.
Prueba	Las pruebas son fundamentales ya que es necesario saber de qué prueba son las calificaciones que vamos a manejar
Estudiante_Prueba	Esta nos muestra los estudiantes que han realizado una prueba y cual fue.
Resultado_Componente	En esta se encuentran los resultados de un estudiante para un componente en una prueba específica.
Resultado_Competencia	En esta se encuentran los resultados de un estudiante para una competencia en una prueba específica.
Resultado_área	En esta se encuentran los resultados de un estudiante para un área

	en una prueba específica.
Periodo Lectivos	Los periodos lectivos son nuestro factor tiempo indispensable en cualquier datamart.

(Tabla 5)

Detalles de las dimensiones

Nombre del atributo	Descripción del atributo	Ejemplo de valores
Id_Estudiante	Es el atributo que se auto incrementa y va a ser el código del estudiante dentro del datamart	1 2 4585
Cod_Estudiante	Este atributo es el código como un estudiante se identifica dentro de la institución.	200857469 201245865 200115468
Nombres	Es el nombre(s) del estudiante	Mauricio Mauricio Carlos
Apellidos	Los apellidos de los estudiantes	Vargas Vargas Gutiérrez Agudelo
Fecha_Nacimiento	Es el año, mes y día en que nació el estudiante.	1993/10/10 1985/05/16

Nombre del atributo	Descripción del atributo	Ejemplo de valores
Id_Componente	Es el atributo que se auto incrementa y va a ser el identificador del componente dentro del datamart	1 2 4585
Componente	Este atributo es el nombre que identifica el componente.	Pragmático Que es el hombre

Nombre del atributo	Descripción del atributo	Ejemplo de valores
Id_Competencia	Es el atributo que se auto incrementa y va a ser el identificador de la competencia dentro del datamart	1 2 4585 2508
Competencia	Este atributo es el nombre que identifica la competencia.	Lectora Escritora

Nombre del atributo	Descripción del atributo	Ejemplo de valores
Id_Área	Es el atributo que se auto incrementa y va a ser el identificador del área dentro del datamart	1 2 4585
Área	Este atributo es el nombre que identifica el área.	Lenguaje Biología
Nombre del atributo	Descripción del atributo	Ejemplo de valores

Nombre del atributo	Descripción del atributo	Ejemplo de valores
Id_Prueba	Es el atributo que se auto incrementa y va a ser el identificador de la prueba dentro del datamart	1 2 4585
Nombre	Este atributo es el nombre que identifica la prueba.	Prueba 1 Prueba para 5°
Id_Nivel_Prueba	Es el atributo que se auto incrementa y va a ser el identificador del nivel dentro del datamart	1 2 3
Nivel_Prueba	Es el atributo que indica que nivel tiene la prueba	3 5

Nombre del atributo	Descripción del atributo	Ejemplo de valores
Id_Resultado_Componente	Es el atributo que se auto incrementa y va a ser el identificador del resultado dentro del datamart	1 2 4585
Id_Componente	La llave foránea que se comunica con el id del componente	1 25
Id_EstudianteXprueba	La llave foránea que se comunica con la dimensión estudiante_prueba	2 1 45
Puntaje	Es el puntaje obtenido por un estudiante para dicho componente en una prueba específica.	3 5 9.2

Nombre del atributo	Descripción del atributo	Ejemplo de valores
Id_EstudianteXprueba	Es el atributo que se auto incrementa y va a ser el identificador de cada tupla de la dimensión estudiantexprueba dentro del datamart	1 2 4585
Id_Estudiante	La llave foránea que se comunica con el id del estudiante	1 25
Id_Prueba	La llave foránea que se comunica con la dimensión Prueba	2 1 45

Nombre del atributo	Descripción del atributo	Ejemplo de valores
Id_Resultado_Componente	Es el atributo que se auto incrementa y va a ser el identificador del resultado dentro del datamart	1 2 4585
Id_Competencia	La llave foránea que se comunica con el id de la competencia	1 25
Id_EstudianteXprueba	La llave foránea que se comunica con la dimensión estudiante_prueba	2 1

Nombre del atributo	Descripción del atributo	Ejemplo de valores
		45
Puntaje	Es el puntaje obtenido por un estudiante para dicho componente en una prueba específica.	3 5 9.2

Nombre del atributo	Descripción del atributo	Ejemplo de valores
Id_Resultado_Área	Es el atributo que se auto incrementa y va a ser el identificador del resultado dentro del datamart	1 2 4585
Id_Área	La llave foránea que se comunica con el id del área	1 25
Id_EstudianteXprueba	La llave foránea que se comunica con la dimensión estudiante_prueba	2 1 45
Puntaje	Es el puntaje obtenido por un estudiante para dicho componente en una prueba específica.	3 5 9.2

Nombre del atributo	Descripción del atributo	Ejemplo de valores
Id_periodo_lectivo	Es el atributo que se auto incrementa y va a ser el identificador del periodo dentro del datamart	1 2 4585
Año_lectivo	Este atributo identifica el nombre que se le asigna al periodo lectivo	2011 2012

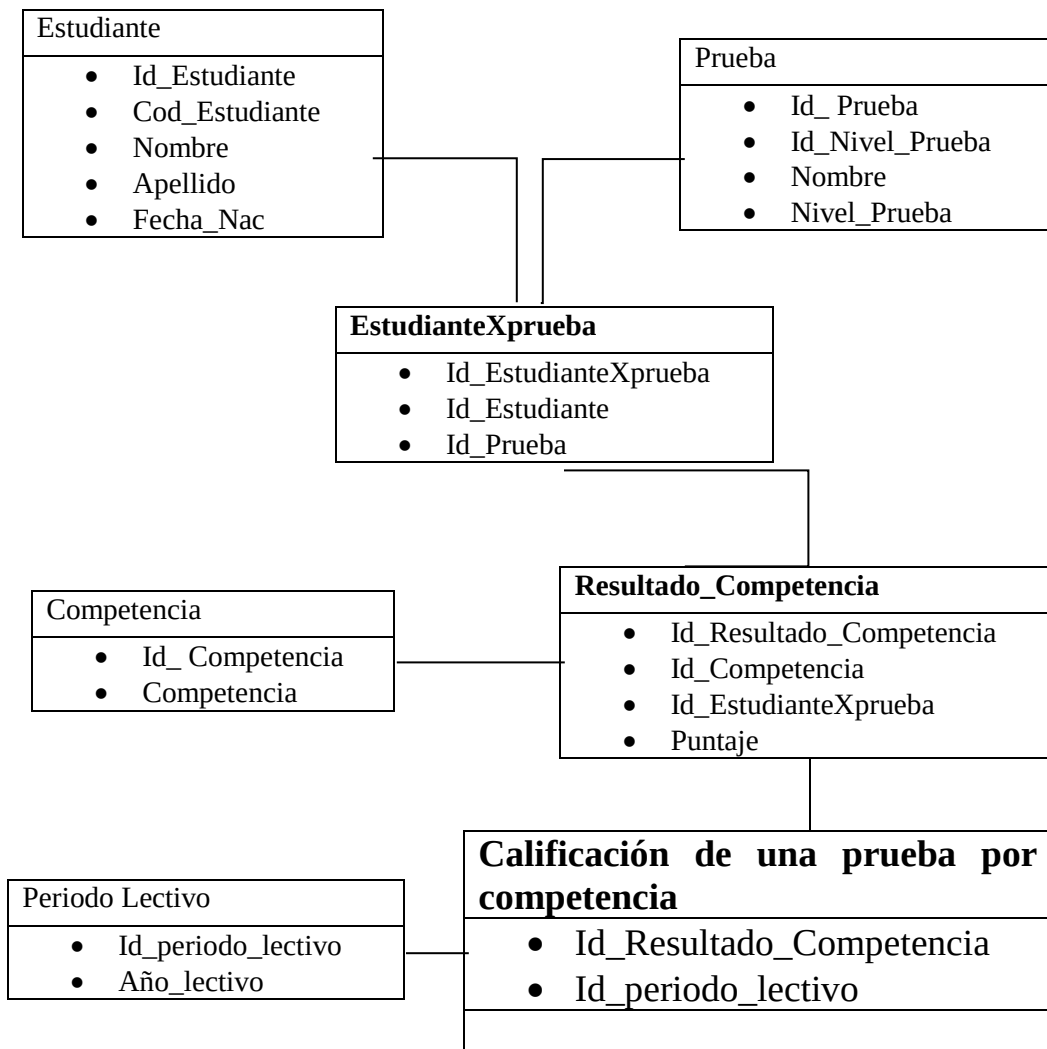
Descripción de los hechos

Nombre del hecho	Descripción del hecho
Calificación de un área	Se le asigna una nota a un estudiante para un área durante un periodo lectivo.
Calificación del componente de una prueba	Se le asigna a un estudiante el puntaje obtenido durante una prueba en cierto periodo lectivo para un componente.
Calificación de la competencia de una prueba	Se le asigna a un estudiante el puntaje obtenido durante una prueba en cierto periodo lectivo para una competencia.

(Tabla 6)

Calificación de la competencia de una prueba

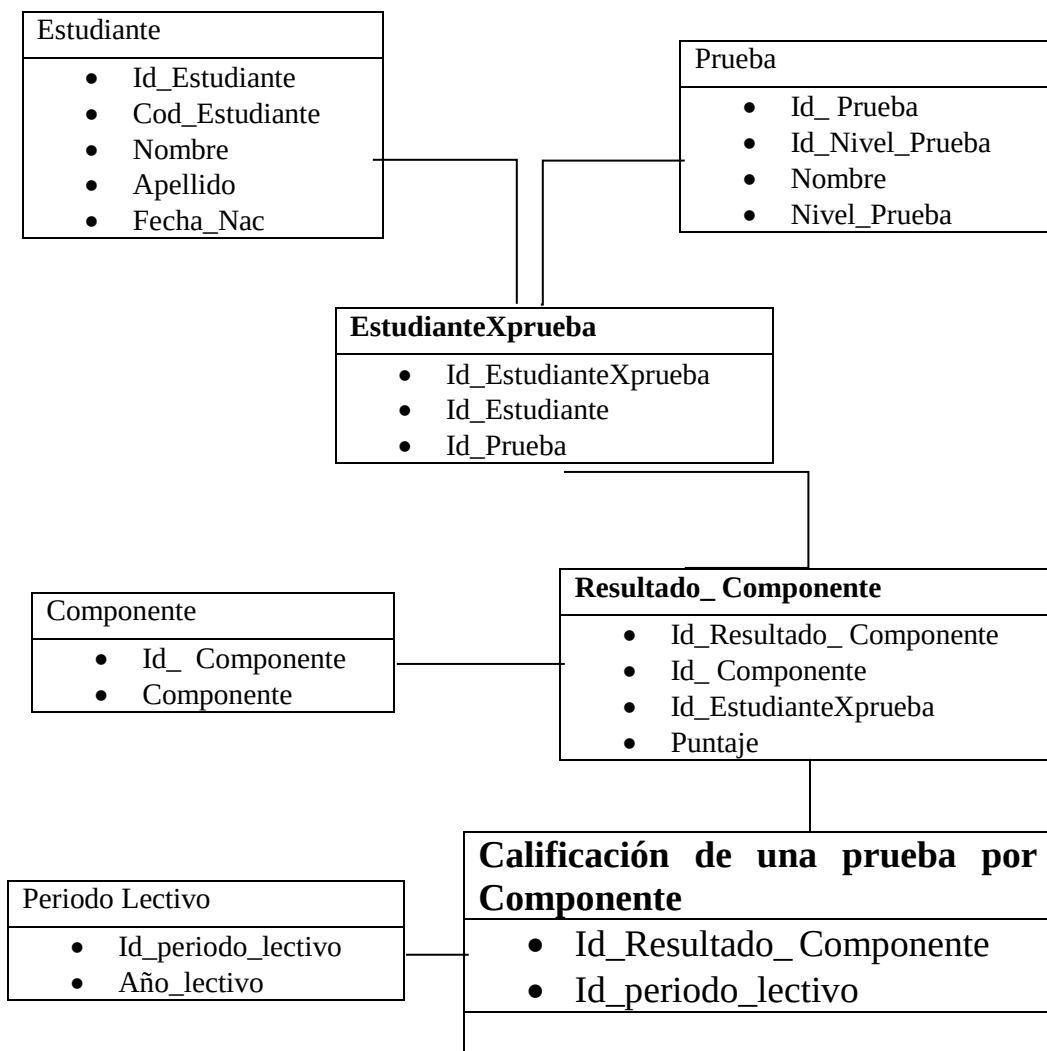
En la figura 14 se muestra la estrella correspondiente al hecho de calificar una competencia durante una prueba.



(Figura 12)

Calificación del componente de una prueba

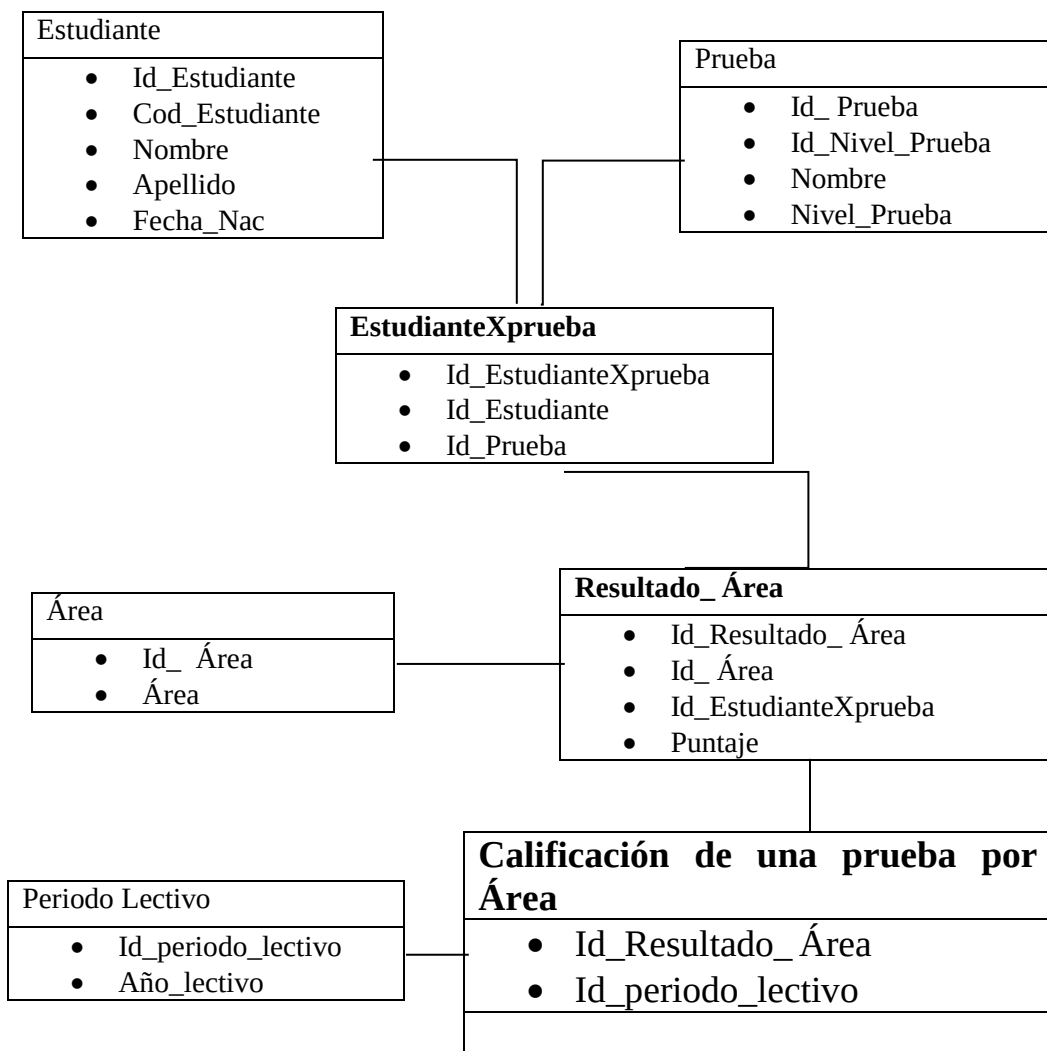
En la figura 13 se muestra la estrella correspondiente al hecho de calificar un componente durante una prueba.



(Figura 13)

Calificación de un área

En la figura 12 se muestra la estrella correspondiente al hecho de calificar un área académica durante una prueba.



(Figura 14)

10.3.2 Carga y Generación de los datos

10.3.2.1 Carga de los datos.

Los estudiantes y los resultados de los componentes, competencias y áreas se cargan desde un archivo en Excel (.xlsx), el sistema almacena todos los datos correspondientes para su posterior utilización en los reportes.

10.3.2.2 Generación de datos

Se obtuvieron datos reales de la Institución Educativa Sagrado Corazón Jesús de una prueba realizada a un grupo de último año por una empresa particular, estos datos que nos fueron entregados tenían los resultados de cada estudiante en cada área, componente y competencia para los cuarenta y cuatro (44) estudiantes.

Con los datos suministrados se estableció los resultados completos para un año lectivo, en base a estos, la media y desviación estándar nacional, la cual fue entregada en este mismo estudio se generaron tres (3) años más con resultados pseudo-aleatorios mediante una distribución normal de los datos, posteriormente de la generación de estos datos se cargaron a la aplicación para alimentar el datamart.

10.3.3 Reportes

Esta aplicación genera dos reportes diferentes usando minería de datos, el primero se denomina análisis por asociación de hechos en el cual el usuario ingresa un componente, competencia o área y la aplicación retorna el porcentaje de estudiantes que consigue un buen resultado en el componente, competencia o área seleccionada con cada uno de los componentes, competencias y áreas que se encuentren en la aplicación.

El segundo se denomina análisis de asociación a priori en el cual el usuario establece el nivel, componente evaluativo, área, el rango en el cual desea el reporte y el periodo del año en el que lo desea, además de esto configura el mínimo soporte (minsup) que es la cantidad de veces que debe aparecer una asociación para ser considerada como importante, igualmente se define el máximo puntaje del área académica que indica desde que puntaje se considera que a un estudiante le fue bien en el área académica.

10.3.3.1 Análisis por asociación

Un ejemplo de este reporte seria:

Dados los siguientes resultados de varias pruebas en diferentes años.

			Componentes			Competencias		
Prueba	Estudiantes	Áreas	1	2	3	1	2	3
Prueba	Estudiante 1	Área 1	8	6.3	4	7.1	4	4.5
		Área 2	7.2	6	5	4	3	7.6
	Estudiante 2	Área 1	9	8	4.6	7.5	1	0
		Área 2	5	6	3.8	6.5	2	3.2
	Estudiante 3	Área 1	6.1	4.9	2.8	7.6	9.1	10

		Área 2	2.6	4.6	9.3	5.6	4.3	2.6
--	--	--------	-----	-----	-----	-----	-----	-----

Mediante las estadísticas que el sistema entrega se ha notado que la institución se encuentra débil en el componente 1 del área 1, la aplicación retornaría los siguientes resultados:

Comparación del componente 1 con el componente 2 del área1, estableciendo como mínimo de nota para ir bien el promedio ósea cinco (5).

Estudiantes	Áreas	1	2
Estudiante 1	Área 1	Bien	Bien
	Área 2	Bien	Bien
Estudiante 2	Área 1	Bien	Bien
	Área 2	Bien	Bien
Estudiante 3	Área 1	Bien	Mal
	Área 2	Mal	Mal

El 100% de las veces que el resultado del componente 2 es bueno el del componente 1 también lo es.

Dado esto el administrador podría tomar la decisión de intensificar el trabajo no solo en el componente 1 sino también en el componente 2, para así lograr una mejoría en el componente débil (en este caso el componente 1).

En la figura 15 se muestra la forma en la cual la aplicación retorna este reporte.

ANÁLISIS POR COMPONENTE			ANÁLISIS POR COMPETENCIA	ANÁLISIS POR ÁREA
Seleccione el componente >>				
Numerico-variacional				
Resultados >>				
Lista de competencias				
N	Competencia	Porcentaje casos buenos		
1	Indagación	69.39%		
2	Explicación de fenómenos	55.1%		
3	Escritora	27.7%		
4	Lectora	27.04%		
5	Uso comprensivo del conocimiento científico	22.43%		
6	Solución de problemas	13.61%		
7	Identificar	12.52%		
8	Razonamiento	12.28%		
9	Comunicación	17%		

(Figura 15)

10.3.3.1 Análisis por a priori

Un ejemplo de este reporte seria:

10.4 Estadísticas

La aplicación ofrece tres tipos de estadísticas, el primero es las estadísticas de área en estas el usuario decide el grado, área, curso, componente evaluativo, el rango y además de si desea que las estadísticas sean generadas para un periodo lectivo, todos los periodos o el promedio del año, además de esto puede agregar un componente o una competencia para que las estadísticas se generen en torno a esto.

El segundo tipo son las estadísticas de profesor en estas el usuario configura exactamente lo mismo que la anterior, pero en esta agrega el profesor sobre el que desea se genere el reporte.

El tercero son las estadísticas de estudiante, en estas se configuran igual que las anteriores pero en este caso no se configura un rango si no un solo periodo lectivo.

10.4.1 Diagramas de barras

Los diagramas de barras se utilizan en esta aplicación para los reportes estadísticos de estudiantes y profesor.

En la figura 16 se muestra un diagrama de barras generado por la aplicación.



Este diagrama muestra un promedio realizado para un área en todos los periodos en el mes de febrero de 2013 por lo que solo en este año se genera barra para el primer periodo.

Aquí se pueden ver el promedio ponderado de todos los resultados de un área para cierto curso; estas estadísticas son de vital importancia para la toma de decisiones, ya que se puede notar claramente los avances y/o retrocesos que se han tenido históricamente en dicho curso.

10.4.2 Diagrama de torta

La aplicación genera este tipo de diagramas en las estadísticas de estudiante ya que en estas estadísticas no se tabulan promedios sino porcentajes.

11 Implementación

En la implementación del sistema se requirió la ayuda de tecnologías de punta y libres para el desarrollo y enriquecimiento de aplicaciones web tales como PHP, AJAX, CSS, Flash y JQuery, y sin ayuda de ningún clase de herramienta para la agilización de la programación de módulos CRUD (Crear, Visualizar, Modificar y Eliminar) e interfaces de usuario como lo son los frameworks.

Para la implementación de un editor de texto requerido dentro de la aplicación para el ingreso de preguntas y enunciados de forma intuitiva y con las herramientas que permiten aplicarle ciertas características al texto ingresado (tamaños, fuentes, alineaciones e incorporación de imágenes), se utilizó la librería YUI Text Editor implementada en javascript.

La generación de gráficas estadísticas es soportada por una librería implementada en flash que mediante la configuración de sus parámetros en formato xml permite crear diferentes tipos de gráficas (pastel, barras, puntos, velas) en 2D y en 3D, de esta manera no solo permite una mayor dinamismo en la implementación sino también un entorno de gráficas agradable al usuario.

Como motor de base de datos se utilizó postgresSQL, herramienta libre y útil en los casos en que se requiere manejar grandes volúmenes de datos, y que además posee una interfaz de usuario muy funcional e intuitiva.

Las tecnologías anteriormente nombradas JQuery y CSS fueron de gran ayuda para la implementación de un ambiente gráfico intuitivo, agradable y llamativo, requerimiento establecido como esencial para un tipo de sistema que va ser utilizado por la comunidad estudiantil y que basa su interés en las aplicaciones por sus gráficas y usabilidad.

Para la carga de información en masa de estudiantes y resultados de pruebas se planteó el uso de archivos de Excel(.xlsx), por lo cual se requirió implementar dentro de la aplicación la librería SimpleXML, la cual lee archivos de este tipo y facilita la información de estos para su uso a través de PHP.

Se obtuvieron resultados de pruebas pre-ICFES realizados en una institución, además de la media y la desviación estándar nacional, en base a esto y mediante la fórmula estadística de la distribución normal se generaron datos pseudo-aleatorios con los cuales se alimentó la base de datos que posteriormente se utilizó para el descubrimiento del conocimiento.

12. CONCLUSIONES

- 1.** Cuando se desarrollan soluciones informáticas para una institución real, la obligación de que la solución se ajuste a las necesidades del cliente y no el cliente a la aplicación, el tiempo de la elección de las tecnologías de información a utilizar suele ser muy extenso, afectando los tiempos presupuestados inicialmente.
- 2.** El desarrollo de una aplicación de gran robustez que cumple con un gran número de requisitos, permitió adquirir experiencia en el diseño e implementación web, así también en el diseño y construcción de un datamart y su posterior utilización para el descubrimiento del conocimiento.
- 3.** Se utilizó un ambiente de trabajo flexible durante todo el proceso, con lineamientos y documentación definida pero maleable, de lo que se puede concluir que una aplicación robusta puede ser realizada con métodos ágiles. En este caso la implementación de una aplicación educativa robusta no conllevó a que la metodología utilizada debiera ser robusta.
- 4.** El desarrollo de una aplicación que necesita de una tecnología de la cual el desarrollador no es un gran conocedor, requiere tiempo de investigación que si no se tiene programado desde el inicio generará demoras en la entrega de la implementación final; para este caso el descubrimiento del conocimiento no era una tecnología con la cual tuviéramos mucha experiencia, lo que nos obligó a investigar muy a fondo este vertiente de las ciencias de la computación.

13. TRABAJOS FUTUROS

1. El sistema funciona solo para los resultados obtenidos en las pruebas pre-icfes, y las técnicas de minerías se realizan sobre estos; se debe crear un módulo en el cual el usuario tenga la posibilidad de cargar al sistema las calificaciones obtenidas por un estudiante en un área durante un periodo lectivo, información necesaria para descubrir nuevo conocimiento en asociación con los datos de resultados arrojados por la aplicación.
2. En una institución se realiza una serie de procesos los cuales tienen un sin número de formatos y en la mayoría de las instituciones se hacen de forma manual; así que ampliar el sistema para que estos formatos se encuentren de manera virtual sería de gran ayuda para las instituciones.
3. Partiendo de que la aplicación desarrollada busca resolver algunos problemas institucionales, un trabajo futuro que aporte al objetivo global de resolver este tipo de problemas, sería la implementación de otros sistemas necesarios en ciertas áreas administrativas de las instituciones.
4. Las instituciones tienen una serie de datos demográficos de los estudiantes los cuales pueden ser utilizados para descubrir conocimiento y posteriormente poder clasificar los estudiantes que sean potencialmente propensos a tener problemas académicos y así ayudar a la institución a implementar estrategias para prevenir dichos problemas.
5. El manejo contable de la institución por obligación debe manejarse de manera sistematizada, así que lograr que el sistema se integre con el sistema contable lo haría mucho más robusto y una herramienta muy efectiva para una institución.

14. BIBLIOGRAFÍA

- [1] Flores M, u_dl_a/tales/documentos/lad/flores_m_a/capitulo1, Disponible en catarina.udlap.mx.
- [2]Mendoza A, Como plantear un proyecto de investigación- Capitulo1, Disponible en <http://tesisyalgomias.blogspot.com>.
- [3]Chaparro Lopez G., Forrero Sarmiento L., Sistemas de información para administración de proyectos de grado, Disponible en <http://www.javeriana.edu.co>.
- [4]Galviz Quintero J., diseño e implementación de una aplicación web para la gestión de las reservas de aulas y videobeams de la universidad de santander udes sede cúcuta, disponible en <http://www.camayih.org>.
- [5]Moody D., Kortink M., From Enterprise Models to Dimensional Models: A Methodology for Data Warehouse and Data Mart Design, 2000.
- [6]Inmon W. H.. Building the Data Warehouse. Third Edition, 2003.
- [7]Fayyad, U.M.; Piatetsky-Shapiro, G.; Smyth, P.; Uthurusamy, R. , Advances in Knowledge and Data Mining. MIT Press. Massachussets, USA, 1996.
- [8]Urdaneta, Elymir. El Data Mining. Universidad de Caracas. Venezuela, (2001).
- [9]Fernández V., Desarrollo de sistemas de información una metodología basada en el modelado. 2010
- [10]Chapman P, Clinton J, Kerber R, CRISP-DM 1.0 step-by-step data mining guide, 2000.
- [11] Pirolo S., Métodos Ágiles , 2009.
- [12] Surhone L.M., Tennoe M.T., Henssonow S.F.,Iconix,2010.