



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**SAK'A: DESARROLLO DE UN ASISTENTE DE ESTUDIO BASADO
EN *WEB 2.0* Y *WEB SEMÁNTICA***

**Luis Antonio Pusey Alvarado
Johan Steve Cristales Mendizábal
César Andrés García Roldán**

Asesorado por el Ing. Javier Gramajo López

Guatemala, junio de 2009

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**SAK'A: DESARROLLO DE UN ASISTENTE DE ESTUDIO BASADO
EN *WEB 2.0* Y *WEB SEMÁNTICA***

TRABAJO DE GRADUACIÓN
PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

LUIS ANTONIO PUSEY ALVARADO
JOHAN STEVE CRISTALES MENDIZÁBAL
CÉSAR ANDRÉS GARCÍA ROLDÁN
ASESORADO POR EL ING. JAVIER GRAMAJO LÓPEZ
AL CONFERÍRSELE EL TÍTULO DE
INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, JUNIO DE 2009

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	Inga. Glenda Patricia García Soria
VOCAL II	Inga. Alba Guerrero de López
VOCAL III	Ing. Miguel Ángel Dávila Calderón
VOCAL IV	Br. José Milton De León Bran
VOCAL V	Br. Isaac Sultán Mejía
SECRETARIA	Inga. Marcia Ivónne Véliz Vargas

TRIBUNAL QUE PRÁCTICO EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Murphy Olympo Paiz Recinos
EXAMINADOR	Ing. Marlon Antonio Pérez Turk
EXAMINADORA	Inga. Floriza Ávila Pesquera
EXAMINADORA	Inga. Sonia Yolanda Castañeda Ramírez
SECRETARIA	Inga. Marcia Ivónne Véliz Vargas

HONORABLE TRIBUNAL EXAMINADOR

Cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración nuestro trabajo de graduación titulado:

SAK'A: DESARROLLO DE UN ASISTENTE DE ESTUDIO BASADO EN *WEB 2.0* Y *WEB SEMÁNTICA*,

tema que nos fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, en febrero de 2008.

Luis Antonio Pusey Alvarado

Johan Steve Cristales Mendizábal

César Andrés García Roldán

ACTO QUE DEDICO A:

Dios, por haber sido la luz que iluminó mi camino, la fuerza que me mantuvo en los momentos de debilidad y la sabiduría en los momentos de duda.

Mi madre, por creer siempre en mi persona, por sus cuidados y el amor incondicional que me ha brindado a lo largo de la vida.

Mi padre, por brindarme oportunidades a lo largo de mi vida.

Mis hermanos, por su amistad, su cariño y por ser mi inspiración para ser una mejor persona cada día.

Mi novia, Fabiola Herrera, por brindarme su amor y compartir esta alegría conmigo.

Mis amigos, por brindarme su amistad, por compartir todos los buenos y malos momentos que pasamos durante la carrera.

La familia Cristales, por brindarme su amistad, apoyo y confianza.

Mis catedráticos, en especial a nuestro asesor y amigo, Javier Gramajo, por sus consejos y por brindarnos su apoyo durante todo el proyecto.

Al Director de la Escuela de Sistemas, Ing. Marlon Turk, por su apoyo en la etapa final del proyecto.

La Gloriosa Universidad de San Carlos de Guatemala, por brindarme no solo conocimientos, sino una escuela para la vida y a personas maravillosas que tengo el gusto de llamarlos amigos.

Luis Antonio Pusey Alvarado

Dios Todopoderoso y a la Virgen María, por ser mi guía, llenarme de tantas bendiciones y haberme permitido alcanzar esta meta.

Mi Padre Edwin Leonel Cristales Archila, por darme todo su apoyo y con su ejemplo la fortaleza de salir adelante para obtener este triunfo.

Mi madre Lissette Vianey Mendizábal Saravia de Cristales, persona única en el mundo, que quiero y amo mucho, estuvo siempre a mi lado y su gran amor es mi motivación a seguir, por eso hoy le entrego este éxito.

Mis hermanos Brian Derick y Jeremy Andreé, por hacerme la vida más feliz con su compañía, compartimos sangre, recuerdos y cariño, también compartiremos nuestros logros.

Mi novia Melany Murillo, por el amor que le tengo y su cariño, apoyo y comprensión siempre que lo necesité.

Mi familia, que la admiro y quiero mucho por haber estado pendientes siempre de mí, por todos sus consejos, paciencia y cariño.

Mis amigos, que tantos buenos recuerdos hemos compartido y que de alguna manera u otra ayudaron a que alcanzara mi meta.

Johan Steve Cristales Mendizábal

Dios y a la Virgen María, quienes fueron los que me dieron fuerza, sabiduría en este proyecto, además me ayudaron a ser perseverante en esta etapa de mi vida.

Mis padres y hermanos que fueron los que me apoyaron en todo momento a lo largo de mi carrera.

Mis compañeros de EPS y al ingeniero Javier Gramajo, ya que cada uno colaboró de forma significativa a que el proyecto se llevara a cabo, cumpliendo así nuestras metas y objetivos trazados.

César Andrés García Roldán

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES	V
GLOSARIO	VII
RESUMEN	XIII
OBJETIVOS.....	XV
INTRODUCCIÓN.....	XVII
1 WEB 2.0	1
1.1 Definición	1
1.2 La <i>web</i> como plataforma.....	3
1.3 La <i>Web 2.0</i> y la educación.....	6
1.4 Herramientas <i>Web 2.0</i> de uso educativo	7
2 DEFINICIÓN DEL PROBLEMA: SOBRECARGA ACADÉMICA . 9	9
2.1 Definición	9
2.2 Herramientas para solucionar el problema	9
2.3 Ontologías.....	10
2.3.1 Tipos de ontologías.....	11
2.3.2 Aplicaciones de las ontologías	12
2.4 <i>Web</i> semántica	13
2.5 Agente inteligente	14
2.6 Descripción de las tecnologías a utilizar	15
2.6.1 XML.....	15
2.6.2 JSF.....	15
2.6.3 <i>Hibernate</i>	16
2.6.4 J2EE	16
3 ESTUDIO DE FACTIBILIDAD.....	17
3.1 Análisis FODA.....	17
3.2 Diagrama FODA	18
3.3 Descripción de la encuesta	19
3.3.1 Análisis de la encuesta	20

3.3.2	Comentarios sobre los resultados de las encuestas.....	27
4	REQUERIMIENTOS DEL SISTEMA.....	29
4.1	Administrar grupos.....	29
4.2	Administrar horarios.....	31
4.3	Administrar eventos.....	32
4.4	Notificar eventos.....	34
4.5	Administrar documentos.....	35
4.6	Administrar enlaces.....	36
4.7	Administrar cuestionarios.....	38
4.8	Contestar cuestionario.....	40
4.9	Consultar estadísticas.....	41
4.10	Administrar mensajes.....	42
4.11	Administrar cuenta.....	44
4.12	Administrar <i>banners</i>	45
4.13	Administrar noticias.....	45
4.14	Administrar contactos.....	46
4.15	Administrar foro.....	48
4.16	Administrar integrantes del grupo.....	49
4.17	Buscar información.....	50
5	MODELO DE DISEÑO DETALLADO DEL SISTEMA.....	53
5.1	Metas y restricciones arquitectónicas.....	53
5.2	Vista de casos de uso.....	54
5.2.1	Diagramas de casos de uso.....	54
5.3	Vista lógica.....	60
5.3.1	Diagrama de paquetes.....	61
5.3.2	Diagrama de clases.....	63
5.4	Vista física.....	67
5.4.1	Diagrama de despliegue.....	68
5.4.2	Diagrama de red.....	68
5.5	Vista de implementación.....	69
5.5.1	Interfaz usuario – presentación.....	69

5.5.2	Controlador	70
5.5.3	Modelos de negocio	71
5.5.4	Datos.....	72
5.5.5	Diagrama de componentes	73
CONCLUSIONES.....		75
RECOMENDACIONES		77
BIBLIOGRAFÍA.....		79

ÍNDICE DE ILUSTRACIONES

FIGURAS

1	Mapa visual de la <i>Web 2.0</i> .	3
2	Descripción de la pregunta número 1 y su resultado.	20
3	Descripción de la pregunta número 2 y su resultado.	21
4	Descripción de la pregunta número 3 y su resultado.	21
5	Descripción de la pregunta número 4 y su resultado.	22
6	Descripción de la pregunta número 5 y su resultado.	23
7	Descripción de la pregunta número 6 y su resultado.	23
8	Descripción de la pregunta número 7 y su resultado.	24
9	Descripción de la pregunta número 8 y su resultado.	24
10	Descripción de la pregunta número 9 y su resultado.	25
11	Descripción de la pregunta número 10 y su resultado.	25
12	Descripción de la pregunta número 11 y su resultado.	26
13	Descripción de la pregunta número 12 y su resultado.	26
14	Diagrama del caso de uso grupos.	55
15	Diagrama del caso de uso eventos.	56
16	Diagrama del caso de uso administrar información.	57
17	Diagrama del caso de uso cuestionarios.	58
18	Diagrama del caso de uso mensajes.	59
19	Diagrama del caso de uso administrar foro.	60
20	Diagrama de paquetes.	63
21	Diagrama de clases (vista).	64
22	Diagrama de clases (modelo).	65
23	Diagrama de clases (controlador).	66
24	Diagrama de despliegue.	68
25	Diagrama de red.	69

26	Diagrama del patrón MVC.	72
27	Diagrama de componentes.	74

TABLA

I	Diagrama FODA.	18
---	----------------	----

GLOSARIO

Actor	Es un conjunto de individuos, computadoras o sistemas que interactúan con el sistema.
Agentes	Es una entidad <i>software</i> que recoge, filtra y procesa información contenida en la <i>web</i> ¹ .
AJAX	<i>Asynchronous JavaScript And XML</i> , es una técnica de desarrollo <i>web</i> para crear aplicaciones interactivas ² .
Arquitectura del Software	Es una serie de decisiones significativas acerca del conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para la construcción del <i>software</i> ³ .
Banner	Imagen, gráfico o texto utilizado con fines publicitarios la cual generalmente presenta un pequeño tamaño, aparece en una página <i>web</i> y habitualmente la enlaza con el sitio <i>web</i> del anunciante ⁴ .

¹ Eduardo Peis Redondo; Enrique Herrera Viedma. **Ontologías, metadatos y agentes: recuperación “semántica de la información”**. Dept. Biblioteconomía y Documentación, Fac. Biblioteconomía y Documentación. Campus de la Cartuja, Universidad de Granada – España.

² Glosario de Informática e *Internet*.
<http://www.panamacom.com/glosario/letra-a.html>.

³ Proyecto Wikipedia. **Arquitectura de Software**.
http://es.wikipedia.org/wiki/Arquitectura_de_software.

⁴ Glosario de Informática e *Internet*.
<http://www.panamacom.com/glosario/letra-b.html>.

Blog	Es un sitio <i>web</i> periódicamente actualizado que recopila cronológicamente textos o artículos de uno o varios autores ⁵ .
BO	Objetos que serán guardados en la base de datos y que hacen referencia a cada una de las entidades definidas en el modelo de base de datos.
Caso de uso	Es una secuencia de interacciones entre un sistema y alguien o algo que usa alguno de sus servicios ⁶ .
Colaboratorio	Punto de encuentro abierto a académicos, investigadores, estudiantes y público en general interesado en la conformación de espacios de aprendizaje en red, flexible y participativo. Permite a varias personas trabajar juntos en un mismo proyecto, aunque se hallen lejos unos de otros ⁷ .
DAO	Son clases que encapsulan métodos de acceso a datos y que podrán ser accedidos por los servicios para devolver o guardar información directamente a la base de datos
Escenario	Es una descripción formal del flujo de eventos que ocurren durante la ejecución de una instancia de casos de uso.
Flash	Es una tecnología para crear animaciones gráficas vectoriales independientes del navegador y que necesitan poco ancho de banda para mostrarse en los sitios <i>web</i> ⁸ .

⁵ Proyecto Wikipedia. **Blog**. <http://es.wikipedia.org/wiki/Blog>.

⁶ Proyecto Wikipedia. **Caso de Uso**. http://es.wikipedia.org/wiki/Caso_de_uso.

⁷ Proyecto Wikipedia. **Colaboratorio**. <http://es.wikipedia.org/wiki/Colaboratorio>.

⁸ Glosario de Informática e *Internet*.
<http://www.panamacom.com/glosario/letra-f.html>.

Foro	Servicio automatizado de mensajes, a menudo moderado por un propietario, en el cual los suscriptores reciben mensajes dejados por otros suscriptores por un tema dado. Los mensajes se envían por correo electrónico ⁹ .
Hibernate	Es un marco de trabajo para Java que facilita el mapeo de atributos entre una base de datos relacional y el modelo de objetos de una aplicación ¹⁰ .
J2EE	<i>Java 2 Enterprise Edition</i> , es una plataforma de programación para desarrollar y ejecutar aplicaciones elaboradas en lenguaje Java ¹¹ .
JSF	La tecnología <i>Java Server Faces</i> es un marco de trabajo para el desarrollo de <i>interfaces</i> de usuario para aplicaciones java ¹² .
Link	Apuntadores hipertexto que sirven para saltar de una información a otra, o de un servidor <i>web</i> a otro, cuando se navega por <i>Internet</i> ¹³ .

⁹ Glosario de Informática e *Internet*. <http://www.panamacom.com/glosario/letra-f.html>.

¹⁰ Proyecto Wikipedia. **Hibernate**. <http://es.wikipedia.org/wiki/Hibernate>.

¹¹ Proyecto Wikipedia. **Java EE**. <http://es.wikipedia.org/wiki/J2EE>.

¹² Proyecto Wikipedia. **Java Server Faces**.
http://es.wikipedia.org/wiki/JavaServer_Faces.

¹³ Glosario de Informática e *Internet*.
<http://www.panamacom.com/glosario/letra-l.html>.

Modelo	Esquema teórico, generalmente en forma matemática, de un sistema o de una realidad compleja, que se elabora para facilitar su comprensión y su estudio ¹⁴ .
Ontologías	Es una descripción formal de los conceptos y las relaciones entre conceptos ¹⁵ .
Open source	Es un término que se aplica a los programas cuyo código fuente está disponible al público ¹⁶ .
Servicio	Clases que contienen métodos que podrán ser accedidos por medio de la capa de presentación para poder obtener o guardar datos.
TIC	Son las tecnologías que permiten difundir, transmitir y procesar información por medios electrónicos de manera instantánea.
Web	Red informática para el intercambio de texto, gráficos y otros objetos multimedia a través de <i>Internet</i> ¹⁷ .
Wiki	Es un sitio <i>web</i> que puede ser editado por varios usuarios. Los usuarios pueden crear, editar, borrar o modificar el contenido de una página <i>web</i> de una manera fácil y rápida ¹⁸ .

¹⁴ Diccionario. <http://www.desartepaz.org/?cat=32&paged=2>

¹⁵ T. Gruber. ***Toward Principles for the Design of Ontologies Used for Knowledge Sharing***. (Technical Report KSL-93-04, Knowledge Systems Laboratory, Stanford University, CA, 1993).

¹⁶ Glosario de Informática e *Internet*.
<http://www.panamacom.com/glosario/letra-o.html>.

¹⁷ La *Web* de un vistazo. <http://www.learnthenet.com/spanish/web/010www.htm>.

¹⁸ Glosario de Informática e *Internet*.
<http://www.panamacom.com/glosario/letra-w.html>.

XML

Sus siglas en inglés son *Extensible Markup Language* (lenguaje de marcas extensible), es un metalenguaje estructurado por marcas o etiquetas desarrollado por el *World Wide Web Consortium* (en adelante W3C)¹⁹.

¹⁹ Proyecto Wikipedia. **XML**. <http://es.wikipedia.org/wiki/XML>.

RESUMEN

La *web* es un medio joven y en constante evolución, ha pasado de ser una *web* estática con usuarios pasivos a una *web* dinámica con usuarios participativos que crean y comparten contenidos. Esta nueva *web* se ha llamado *Web 2.0* o *Web Social* que tiene como principal componente al usuario.

La filosofía de la *Web 2.0* se basa en el conocimiento social. Potencia la comunicación, la colaboración y el intercambio de información. Si la docencia se basa en esta misma filosofía sería absurdo no aprovechar la *Web 2.0* para desarrollar en los estudiantes la comunicación y el intercambio de conocimiento colaborativo.

Los profesores se encuentran ante nuevas generaciones de alumnos que son nativos digitales que han nacido y crecido con la *web*, por tal motivo, necesitan desarrollar en sus alumnos competencias de acorde a la sociedad de la información, ya que para estos estudiantes la tecnología nos les parece un ámbito hostil y se adaptan rápidamente a ella, algo que parece muy difícil a las antiguas generaciones que están en un proceso de migración a las nuevas tecnologías.

La información ya no sólo se encuentra en los centros educativos, también está afuera y circula por la *web*, por lo tanto, surge la necesidad de una herramienta que ayude a los estudiantes a clasificar de una forma más eficiente la información existente en la *web*. Gracias a la *Web Semántica*, el *software* tiene la capacidad de procesar el contenido generado por los usuarios, razonar con éste y ejecutar deducciones lógicas.

OBJETIVOS

General

Crear una herramienta que funcione como un asistente de estudio basado en la *Web 2.0* y *Web Semántica*, dirigido a usuarios académicos (estudiantes y profesores), para realizar tareas, proyectos, para facilitar la enseñanza y que encuentren en ella un apoyo para enfrentar la sobrecarga académica, generando, compartiendo y clasificando información de forma más eficiente.

Específicos:

1. Realizar una encuesta que permita conocer las necesidades de los estudiantes para el desarrollo de la herramienta propuesta.
2. Realizar un estudio de factibilidad de la herramienta mediante un análisis de sus fortalezas, oportunidades, amenazas y debilidades.
3. Presentar el concepto de *Web 2.0* y su aplicación en la educación.
4. Desarrollar una ontología en la que se fundamentará la herramienta propuesta, tomando como base lo propuesto en la *Web Semántica* y en la *Web 2.0*.
5. Elaborar la arquitectura de la herramienta con el patrón: Modelo Vista Controlador y representarla con el modelo 4 + 1 vistas.

6. Desarrollar la herramienta con base en la arquitectura propuesta utilizando J2EE como plataforma de programación, *Java Server Faces* para la parte gráfica, *hibernate* para conectar la lógica de la aplicación con la base de datos y MySQL como gestor de la base de datos.

INTRODUCCIÓN

Actualmente, la llegada de las Tecnologías de la Información y de las Comunicaciones (en adelante TICs) a la educación, se ha dado un paso importante en la transformación tanto en los fines como en los métodos y formas del sistema educativo. Por lo que los centros de estudio deben incorporar las TICs como herramienta en su modelo de docencia, investigación, gestión y desarrollo.

La *Web 2.0* ha proporcionado una serie de servicios que permiten acceder y repartir recursos digitales en cualquier forma, para que exista una educación sin complejos de inferioridad, por medio de nuevas tecnologías que ya están al alcance de todos y así formando redes sociales, colaboratorios, *wikis*, *blogs*, etc. Esto permite crear a los estudiantes sus propias oportunidades basadas en el conocimiento generado de forma individual y colectiva.

La sobrecarga académica que experimentan los estudiantes hoy en día crea la necesidad de una herramienta que los ayude a gestionar sus actividades, encontrar y clasificar la información existente. Una herramienta cuyo objetivo es que los estudiantes no trabajen más arduamente por la excelencia, sino más inteligentemente generando conocimiento y beneficiándose del conocimiento que generan otros estudiantes.

En el primer capítulo se define de forma general, cuál es la filosofía de la *Web 2.0*, se presenta la *Web 2.0* como plataforma y las herramientas que ésta tiene para apoyar la educación.

En el segundo capítulo se define el problema que se pretende resolver con el *software* propuesto en un contexto académico, adicionalmente se dan los conceptos básicos sobre las herramientas y tecnologías que se utilizaron para desarrollar el *software*, entre estas herramientas vale la pena resaltar el uso de *ontologías*, que son un conjunto de conceptos relacionados que pertenecen a un dominio, el uso de la *Web Semántica* que se apoya en las ontologías y cuyo objetivo es almacenar la información para que ésta pueda ser interpretada y utilizada por las computadoras sin necesidad de la intervención humana. En la parte final del capítulo se describen brevemente las tecnologías que se utilizarán para el desarrollo del proyecto, entre las cuales están: XML, JSF, *Hibernate* y J2EE.

En el tercer capítulo se hace un estudio de la factibilidad sobre el *software* propuesto en el ámbito académico, para este estudio se publicó una encuesta en *Internet* y fue contestada por estudiantes de la Facultad de Ingeniería en Ciencias y Sistemas de la Universidad de San Carlos de Guatemala, que son los más afectados por la sobrecarga académica. El estudio de factibilidad fue utilizado también como fuente para sacar los requerimientos del *software* que se describen detalladamente en el capítulo cuarto.

El diseño detallado de la arquitectura del *software* se expone en el capítulo cinco, para representar la arquitectura del sistema desde varios puntos de vista, se utilizó el modelo 4 + 1 vista.

1 WEB 2.0

1.1 Definición

El concepto *Web 2.0* debe su origen a una tormenta de ideas entre los equipos de *O'Reilly Media* y *MediaLive International* a mediados de 2004, fortalecido por la primera *Web 2.0 Conference* en octubre de ese mismo año. Con el objeto de dar cierta entidad teórica al nuevo término y contrarrestar la confusión del momento, O'Reilly publicó en septiembre de 2005 lo que hasta hoy es la principal referencia bibliográfica del concepto. Se trata del artículo: *What Is Web 2.0. Design Patterns and Business Models for the Next Generation of software*²⁰.

La *Web 2.0* es la transición que se ha dado de las aplicaciones tradicionales hacia aplicaciones que funcionan a través de la *web* enfocadas al usuario final. Se trata de aplicaciones que generen colaboración y servicios que reemplacen las aplicaciones de escritorio. La *Web 2.0* es más una filosofía que una tecnología²¹.

Algunos de los conceptos que se relacionan con la *Web 2.0* según la *Fundación Orange*²² son (ver figura 1):

²⁰ Cristóbal Cobo Romaní; Hugo Pardo Kuklinski. **Planeta Web 2.0. Inteligencia colectiva o medios fast food.** (Grup de Recerca d'Interaccions Digitals, Universitat de Vic. Flacso México, Barcelona / México DF, 2007) p. 27.

²¹ Educastur. **Web 2.0 y Educación.**
http://blog.educastur.es/files/2007/06/web2_0v02.pdf.

²² Fundación Orange. **Mapa Visual de la Web 2.0.**
<http://www.internality.com/web20/>.

- **Comunicación:** transmitir información en cualquier formato, nivel personal o colectivo de forma simple y fácil, sigue siendo una de las prioridades de los usuarios.
- **Compartir:** frente a los entornos cerrados y la información individual, compartir información en cualquier formato redundante en beneficio para todos.
- **Recomendación:** en un mundo sobresaturado de información, sistemas de filtrado colaborativo y participativo permiten generar recomendaciones fiables.
- **Conversaciones:** actitud de las empresas y organizaciones a relacionarse directamente y de forma transparente con los consumidores gracias a las nuevas formas de comunicación y también internamente.
- **Periodismo ciudadano:** además de consumir información, el usuario escribe *blogs*, toma fotos, graba videos, los comparte, filtra y comenta. Crea un tipo de periodismo más colectivo y directo que el tradicional.
- **Folksonomía:** metodología de clasificación en la que los propios usuarios emplean etiquetas de modo descentralizado sobre objetos diversos tales como fotografías, páginas, videos o textos.

Figura 1. Mapa Visual de la *Web 2.0*



Fuente: Fundación Orange, www.internality.com/web20.

1.2 La *web* como plataforma

Gracias a la *web* muchos servicios dejan de ser aplicaciones encerradas en el ordenador personal para estar disponibles y ser usados vía *web* desde cualquier lugar.

Según el artículo mencionado de O'Really (2005)²³, las aplicaciones *Web 2.0* deben tener estos principios:

²³ Tim, O'Reilly. What is a *Web 2.0*.

<http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>

Aprovechar la inteligencia colectiva: en un entorno como la *web* los usuarios actúan de forma pasiva, navegando a través de los contenidos; o bien en forma activa, creando y aportando a los contenidos. Para ejemplificar este principio podemos citar a Wikipedia que es uno de los productos más representativos de la *Web 2.0*. Wikipedia es una enciclopedia libre, en donde cualquier usuario puede aportar una definición de un término y cualquiera puede corregirlo, transformando al usuario de un consumidor a un co – desarrollador con la plataforma. En este modelo de participación no todo es bueno, el modelo tiene ciertos problemas para los cuales nuevamente se citara a Wikipedia, en donde la estructura sitúa al mismo nivel a escritores principiantes y profesionales. Esto provoca algunas imprecisiones en la información, debido a que no hay filtros en la entrada de la información solo en la salida. La aplicación permite al usuario publicar la información y luego la comunidad decide que tanta relevancia tiene el contenido. Con este modelo el papel del editor tradicional pierde poder en el negocio de la jerarquización y organización de la información, en un ambiente donde ya no solo él decide que poner y que ignorar.

La gestión de la base de datos como competencia básica: lo más valioso de las aplicaciones *Web 2.0* son los datos. El interés en la gestión de la base de datos es obtener información de una masa crítica de usuarios que produce una un volumen datos de gran valor. Poseer esta información es clave sumando los usuarios co – desarrolladores para darle un valor comercial a los productos. Un ejemplo de esto es amazon que obtuvo una política de enriquecimiento de información, de la cual una parte es generada por los usuarios. Hoy en día cuenta con una potente base de datos que es su competencia básica y parte esencial de la riqueza del servicio.

El fin del ciclo de las actualizaciones de versiones de *software*: la *Web* 2.0 rompe con el modelo inicial del *software* con derechos de uso, para pasar al uso del *software* como servicio gratuito, corriendo en la propia *web*. Esto representa un cambio en el modelo de negocio y beneficiar principalmente al usuario. Las aplicaciones *Web* 2.0 desarrollan productos que tienen el objetivo de substituir al *software* empaquetado y al mismo tiempo ganar usuarios que estén dispuestos a colaborar con las nuevas plataformas.

Modelos de programación ligera. Búsqueda de la simplicidad: la forma en la que deben estar diseñados los sitios *web* deben promover la simplicidad y fiabilidad para aplicaciones no centralizadas y escalables. La simplicidad genera formas de reutilizar la programación y la conduce hacia la estandarización. Es importante que siempre se tome en cuenta al usuario como productor de contenidos.

El *software* no limitado a un solo dispositivo: los servicios *Web* 2.0 no se limitan a las computadoras, aunque los primeros servicios fueron pensados solo para estas. Hoy en día los dispositivos móviles se han vuelto en medios para el entretenimiento, gestión de la información y el fortalecimiento de las redes sociales.

Experiencias enriquecedoras para el usuario: con el paso del tiempo la *web* se ha ido transformando cada vez más amigable para el usuario, presentando las interfaces con la capacidad de acceso desde cualquier lugar y momento a los servicios *web*, con la usabilidad y sencillez de los sistemas operativos por medio de herramientas como *flash*, AJAX, etc.

1.3 La Web 2.0 y la educación

La Web 2.0 se ha vuelto una herramienta fundamental en el contexto de la educación aumentando la posibilidad de enseñar a más estudiantes y brindar a los educadores nuevos recursos y enriquecer su proceso de aprendizaje.

La estructura que tenía la antigua web era de pocos emisores y muchos receptores de información, la web evoluciono a una estructura donde hay muchos emisores y muchos receptores, esto ayuda que los estudiantes pueden generar información y a su vez compartirla. Los servicios más populares en el ámbito académico son: *blogs*, *wikis* y colaboratorios.

Según Johnson (1992)²⁴, el conocimiento se genera bajo una serie de negociaciones y no se produce hasta que varios de los intereses de los actores participantes estén incluidos. Propone tres tipos de aprendizaje:

Aprender haciendo (*learning - by - doing*): en este tipo de aprendizaje consiste en que los estudiantes aprendan sobre algún tema generando conocimiento (texto, audio o video) y posteriormente el docente revise el conocimiento generado, haciendo observaciones y corrigiendo los errores. Esta forma de aprendizaje colectivo promueve un aprendizaje constructivo. Las herramientas web que ayudan en este aprendizaje son las que permiten lectura y escritura de información.

Aprender interactuando (*learning - by - interacting*): este aprendizaje se da por medio del intercambio de ideas entre los usuarios de la web. Algunos ejemplos de este tipo de aprendizaje son: *blogs*, correo electrónico y el chat.

²⁴ B. Johnson. *Institutional learning*. (National systems of innovation. Toward a theory of innovation and interactive learning, 1992).

Aprender buscando (*learning - by - searching*): este tipo de aprendizaje se da cuando algún usuario busca información en la *web* para realizar un ensayo, tarea o trabajo escrito. El proceso de investigación, selección y adaptación de la información amplía el conocimiento de quien lo realiza.

1.4 Herramientas *Web 2.0* de uso educativo

Blog: un *blog* es un sitio *web* que funciona como una bitácora en donde los uno o más autores escriben artículos sobre temas de su interés, los cuales son recopilados de forma cronológica y ordenados de forma ascendente. El autor del *blog* deja escrito lo que él considera pertinente sobre el tema.

Wiki: son páginas *web* que tienen enlaces, imágenes y cualquier tipo de contenido que pueda ser visitado y editado por cualquier persona. Esta herramienta permite crear documentos de forma colectiva sin necesidad a que haya un proceso previo de la aceptación del contenido.

Colaboratorio: es un espacio de intercambio colectivo. El objetivo de esta herramienta es la formación de espacios de cooperación recíproca donde el elemento principal es la posibilidad de compartir conocimientos. Aquí se pueden compartir documentos, proyectos, reportes, etc. Tal como si fuese una biblioteca o un laboratorio.

2 DEFINICIÓN DEL PROBLEMA: SOBRECARGA ACADÉMICA

2.1 Definición

La globalización ha alterado los patrones de comunicación y competitividad, encontrando en el conocimiento y la capacitación profesional su principal soporte. Como consecuencia, las universidades se han visto en la tarea de elevar el nivel académico de sus alumnos para que éstos puedan hacer frente a las exigencias del mercado y ser agentes de desarrollo en su país; esto ha provocado una gran sobrecarga académica para estudiantes y personal docente, viéndose reflejado en los trabajos y proyectos elaborados por profesores y estudiantes que no se concluyen o al menos no se realizan con el nivel de calidad del cual ellos son capaces.

Por esta razón, las universidades no pueden aspirar a formar un producto terminado y deben dedicarse a generar profesionales que en su formación adquieran ciertas destrezas, incluyendo buenos hábitos de estudio y planificación de sus actividades, pues ya existe un nuevo tipo de analfabetismo, donde entran aquellas personas que no han aprendido a aprender, reutilizar información de calidad y trabajar eficientemente.

2.2 Herramientas para solucionar el problema

De acuerdo a las herramientas mencionadas en el capítulo 1 y a la definición del problema. Los elementos que se utilizarán para desarrollar la herramienta que ayude a los estudiantes con la sobrecarga académica son:

2.3 Ontologías

Una ontología es una descripción formal de los conceptos y las relaciones entre conceptos²⁵. La definición anterior plantea a las ontologías como un conjunto de conceptos y relaciones que pertenecen a algún dominio.

Las ontologías proveen un entendimiento común de un dominio que permite la comunicación entre las personas y las aplicaciones en sistemas heterogéneos. Una ontología provee una conceptualización que describe la semántica de los datos. Tiene una función similar a un esquema de base de datos. Las ontologías son necesarias por los agentes y por las herramientas de análisis para recuperar información de la *web*²⁶.

Las ontologías tienen los siguientes componentes que servirán para representar el conocimiento de algún dominio²⁷:

- **Conceptos:** son las ideas básicas que se intentan formalizar. Los conceptos pueden ser clases de objetos, métodos, planes, estrategias, procesos de razonamiento, etc.
- **Relaciones:** representan la interacción y enlace entre los conceptos del dominio. Suelen formar la taxonomía del dominio. Por ejemplo: subclase-de, parte-de, parte-exhaustiva-de, conectado-a, etc.

²⁵ T. Gruber. *Toward Principles for the Design of Ontologies Used for Knowledge Sharing*. (Technical Report KSL-93-04, Knowledge Systems Laboratory, Stanford University, CA, 1993).

²⁶ Javier Gramajo. *State of the Art: Ontologies*. (Guatemala, 2005) p. 1-2.

²⁷ T. Gruber. *A Translation Approach to Portable Ontology Specifications, Knowledge Acquisition*. (1993) p. 199-220.

- **Funciones:** son un tipo concreto de relación donde se identifica un elemento, mediante el cálculo de una función que considera varios elementos de la ontología. Por ejemplo, pueden aparecer funciones como categorizar-clase, asignar-fecha, etc.
- **Instancias:** se utilizan para representar objetos determinados de un concepto.
- **Axiomas:** son teoremas que se declaran sobre relaciones que deben cumplir los elementos de la ontología. Por ejemplo: “Si A y B son de la clase C, entonces A no es subclase de B”, “Para todo A que cumpla la condición C1, A es B”, etc.

2.3.1 Tipos de ontologías

Según su dependencia y relación con una tarea específica para un determinado punto de vista, Guarino²⁸ clasifica las ontologías en:

- **Ontologías de alto nivel o genéricas:** están destinadas a describir todos los conceptos generales tales como el espacio, el tiempo, la materia, el objeto, el hecho, la acción, etc.
- **Ontologías de dominio:** describen el vocabulario relacionado con un dominio genérico.
- **Ontologías de tareas:** describen una tarea, actividad o artefacto, por ejemplo componentes, procesos o funciones.
- **Ontologías de aplicación:** describen los conceptos conforme a un campo determinado o unas tareas concretas, que resultan, en muchas ocasiones, especializaciones de diversas ontologías.

²⁸ N. Guarino. *Formal ontology and information systems*. (En GUARINO, N. (ed.), *Formal Ontology in Informa-ton Systems*. Amsterdam: IOS Press, 1998), p. 3-15.

2.3.2 Aplicaciones de las ontologías

Una de las aplicaciones más comunes de las ontologías es la forma como cada persona entiende y representa las cosas que lo rodean. Esto se debe a que el mundo está formado por una serie de conceptos que pertenecen a uno o más dominios que a su vez están relacionados, la mente de cada persona asocia la información que percibe a un dominio y la va clasificando.

Una de las aplicaciones primordiales de las ontologías en el área de informática es favorecer la comunicación entre personas, organizaciones y aplicaciones, para lograr una interoperabilidad entre los sistemas informáticos y los usuarios de estos, porque proporcionan un entendimiento e interpretación común de la información, de forma que se eliminan confusiones conceptuales y terminológicas. Las ontologías resultan muy útiles para facilitar el razonamiento automático, es decir, sin intervención humana. Partiendo de unas reglas de inferencia, un motor de razonamiento puede usar los datos de las ontologías para inferir conclusiones de ellos²⁹.

²⁹ Miguel Ángel Abián. **Ontologías: Qué son y para qué sirven.**

<http://www.wshoy.sidar.org/index.php?2005/12/09/30-ontologias-que-son-y-para-que-sirven>.

2.4 Web semántica

La *Web* semántica es una *web* extendida, dotada de mayor significado en la que cualquier usuario en *Internet* podrá encontrar respuestas a sus preguntas de forma más rápida y sencilla gracias a una información mejor definida. Al dotar a la *web* de más significado y, por lo tanto, de más semántica, se pueden obtener soluciones a problemas habituales en la búsqueda de información gracias a la utilización de una infraestructura común, mediante la cual, es posible compartir, procesar y transferir información de forma sencilla. Esta *web* extendida y basada en el significado, se apoya en lenguajes universales que resuelven los problemas ocasionados por una *web* carente de semántica en la que, en ocasiones, el acceso a la información se convierte en una tarea difícil y frustrante³⁰.

El objetivo de la *Web* semántica es que la información almacenada pueda ser interpretada y utilizada por las computadoras sin la intervención humana, de manera que los agentes *web* puedan ser diseñados para tratar la información que se expone en los diferentes sitios *web* de forma semiautomática. Esto se logra referenciando la información de las páginas *web* a metadatos que están bajo un esquema de datos que pertenecen a un dominio.

Esto ayudará a resolver los problemas de búsqueda de información en *Internet*, permitiendo que los agentes *web* sean capaces de realizar deducciones sobre el contenido de las páginas *web* y la búsqueda de información sea más rápida y precisa.

³⁰ World Wide *Web* Consortium. **Guía breve de la *Web* Semántica.**
<http://www.w3c.es/Divulgacion/Guiasbreves/WebSemantica>.

2.5 Agente inteligente

Un agente es una entidad *software* que recoge, filtra y procesa información contenida en la *web*, realiza inferencias sobre dicha información e interactúa con el entorno sin necesidad de supervisión o control constante por parte del usuario. Estas tareas son realizadas en representación del usuario o de otro agente³¹.

Según Hendler³², las cualidades que debe tener un agente inteligente son:

- **Comunicativo:** el agente debe entender las necesidades, objetivos y preferencias del usuario, ya que de dicha comunicación o interacción usuario-agente depende que el agente pueda llevar a cabo su función de forma eficiente. Así mismo debe poder comunicarse con el entorno mediante representaciones compartidas de conocimiento (ontologías).
- **Autónomo:** el agente, además de comunicarse, debe poder interactuar con el entorno, tomando decisiones y actuando por sí solo, limitando sus acciones según el nivel de autonomía permitida por el usuario.
- **Adaptable:** el agente debe ser capaz de aprender del entorno: usuarios (preferencias), fuentes de información y otros agentes.

³¹ Eduardo Peis Redondo; Enrique Herrera Viedma. **Ontologías, metadatos y agentes: recuperación “semántica de la información”**. Dept. Biblioteconomía y Documentación, Fac. Biblioteconomía y Documentación. Campus de la Cartuja, Universidad de Granada – España.

³² James Hendler. ***Is There an Intelligent Agent in Your Future?***.
<http://www.nature.com/nature/webmatters/agents/agents.html>

2.6 Descripción de las tecnologías a utilizar

2.6.1 XML

Sus siglas en inglés son *Extensible Markup Language* (lenguaje de marcas extensible), es un metalenguaje estructurado por marcas o etiquetas desarrollado por el *World Wide Web Consortium* (en adelante W3C). Es una simplificación y adaptación del SGML y permite definir la gramática de lenguajes específicos. Por lo tanto, XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades³³. El objetivo de XML es estructurar la información de forma estándar, para que esta pueda ser compartida entre sistemas.

2.6.2 JSF

La tecnología *Java Server Faces* es un marco de trabajo para el desarrollo de interfaces de usuario para aplicaciones java. Utiliza *Java Server Pages* (JSP) como la tecnología que permite hacer el despliegue de páginas³⁴.

Según³⁴, sus principales componentes son:

- Un conjunto de APIs para representar componentes de interfaz de usuario para manejar su estado, manejo de eventos, validación del lado del servidor y conversión de datos; definir la navegación entre páginas; soportar internacionalización y accesibilidad; y proporcionar extensibilidad para todas estas características.

³³ Proyecto Wikipedia. **XML**. <http://es.wikipedia.org/wiki/XML>.

³⁴ Proyecto Wikipedia. **Java Server Faces**.
http://es.wikipedia.org/wiki/JavaServer_Faces.

- Un conjunto de componentes por defecto para el desarrollo de interfaces de usuario.
- Dos librerías de etiquetas *Java Server Pages* (JSP) personalizadas para dibujar componentes UI dentro de una página JSP.
- Un manejador de eventos del lado del servidor.
- Un administrador de estados.
- Un administrador de *beans*.

2.6.3 *Hibernate*

Hibernate es un marco de trabajo para Java que facilita el mapeo de los entre los atributos de una base de datos relacional y el modelo de objetos de una aplicación mediante un archivo XML en donde se definen cómo se almacenan, eliminan, actualizan y recuperan los objetos Java³⁵.

2.6.4 J2EE

Java 2 Enterprise Edition, es una plataforma de programación para desarrollar y ejecutar aplicaciones elaboradas en lenguaje Java. Tiene una arquitectura de N capas distribuidas, basándose ampliamente en componentes de *software* modulares ejecutándose en un servidor de aplicaciones³⁶.

³⁵ Proyecto Wikipedia. ***Hibernate***. <http://es.wikipedia.org/wiki/Hibernate>.

³⁶ Proyecto Wikipedia. ***Java EE***. <http://es.wikipedia.org/wiki/J2EE>.

3 ESTUDIO DE FACTIBILIDAD

3.1 Análisis FODA

Fortalezas

- Utilización de herramientas *open source*, esto implica que el *software* no se necesita licencias.
- Información disponible en cualquier momento.
- No se necesitan grandes requerimientos de hardware únicamente un navegador.
- El usuario tendrá un espacio en conjunto con una serie de herramientas para compartir y generar información.

Oportunidades

- Utilización de la *Web 2.0* como plataforma.
- Aprovechamiento de la inteligencia colectiva.
- Surgimiento de los dispositivos móviles con acceso a *Internet*.

Debilidades

- Conexión a *Internet* necesaria para funcionalidad total.
- No hay una entidad oficial que garantice la veracidad de la información.

Amenazas

- La información es manipulada por muchas personas y no se sabe que uso le dan.
- La herramienta esta publicada en *Internet*, lo cual la deja expuesta a ataques de terceras personas hacia la integridad de la misma.

3.2 Diagrama FODA

Tabla I. Diagrama FODA.

	Fortalezas	Debilidades
Oportunidades	La utilización de la <i>web</i> como plataforma, nos permite el acceso a la herramienta únicamente con un navegador y acceso a <i>internet</i> .	Los ataques de personas malintencionadas pueden botar el enlace con el sitio o bien generar información falsa.
Amenazas	Continuamente se deben buscar nuevas herramientas que puedan brindarle protección al sitio contra terceras personas.	Una persona podrá generar información falsa y no habrán mecanismos para detectarla, más que los usuarios.

3.3 Descripción de la encuesta

La encuesta fue publicada en un sitio *web* para que fuera más fácil el acceso a esta. Se tomo una muestra de 100 estudiantes de ingeniería en sistemas de la universidad de San Carlos de Guatemala.

La estructura de la encuesta está formada por tres secciones de las cuales cada una tiene un propósito diferente, la primera sección está formada de 3 preguntas que tienen como objetivo conocer el estado actual de los sistemas de comunicación que ofrece la universidad y el grado de satisfacción de los estudiantes sobre dichos sistemas. La segunda sección consta de 4 preguntas que tienen como objetivo conocer la sobrecarga académica ejercida sobre los encuestados y sus hábitos para planificar sus actividades. Por último la tercera sección consta de 5 preguntas que tienen como objetivo proponer una serie de alternativas para ayudar a las diferentes dificultades que enfrentan los estudiantes diariamente.

La encuesta fue publicada en un sitio en *Internet* para que fuera más cómodo el acceso a esta y tuviera más alcance, el enlace directo hacia la encuesta es:

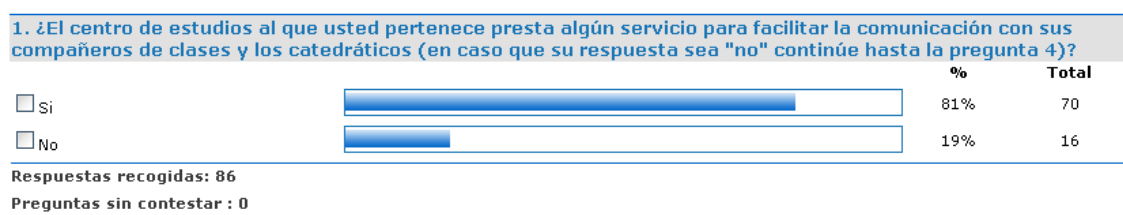
<http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183>

3.3.1 Análisis de la encuesta

Sección 1

En esta sección las preguntas se enfocan para conocer el estado actual de los sistemas de comunicación que brindan la universidad para establecer la comunicación catedrático/estudiante y de la misma manera se pretende conocer el grado de satisfacción del estudiante sobre dicho sistema.

Figura 2. Descripción de la pregunta número 1 y su resultado.





Fuente: Luis A. Pusey, **Proyecto Sak'a,**

<http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183>.

Resultados pregunta 1

En este caso vemos que el 80 % de los estudiantes encuestados conocen el sistema de comunicación con sus catedráticos y compañeros de clases.

Figura 3. Descripción de la pregunta número 2 y su resultado.

2. ¿Esta usted satisfecho con el servicio de comunicación que presta su centro de estudios?			
		%	Total
<input type="checkbox"/> Si		39%	7
<input type="checkbox"/> No		61%	11
Respuestas recogidas: 18			
Preguntas sin contestar : 68			






Fuente: Luis A. Pusey, **Proyecto Sak'a,**

[http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183.](http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183)

Resultados pregunta 2

Más del 60% de los estudiantes encuestados no están satisfechos con el sistema que les brinda su centro de estudios, por lo que surge la necesidad de un nuevo sistema de comunicación.

Figura 4. Descripción de la pregunta número 3 y su resultado.

3. ¿Que servicios presta el sistema de comunicación del centro de estudios al que usted pertenece?			
		%	Total
<input type="checkbox"/> Permite la comunicación directa con el catedrático		62%	10
<input type="checkbox"/> Permite la comunicación directa con el ayudante de cátedra		75%	12
<input type="checkbox"/> Permite la comunicación con los compañeros de clases		31%	5
<input type="checkbox"/> Notifica eventos importantes de su centro de estudios		81%	13
<input type="checkbox"/> El sistema lleva un control de su rendimiento académico		6%	1
Respuestas recogidas: 16			
Preguntas sin contestar : 70			

Fuente: Luis A. Pusey, **Proyecto Sak'a,**

[http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183.](http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183)

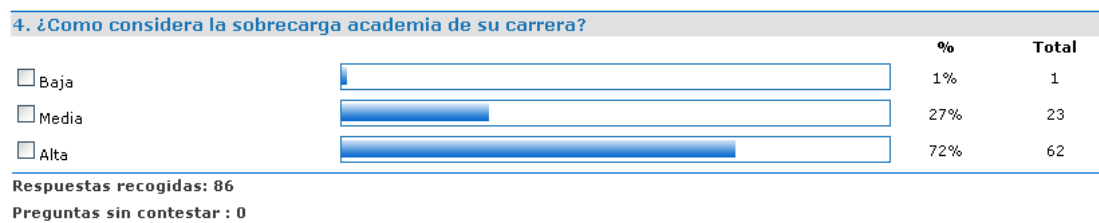
Resultados pregunta 3

Como podemos observar, el objetivo de la pregunta fue obtener el nivel de detalle de los servicios que brinda la universidad, colocando las operaciones básicas que este debe brindar. Podemos observar que si posee los servicios básicos, pero no los necesarios para que los estudiantes puedan agilizar su aprendizaje, comunicación e interacción.

Sección 2

Las preguntas de esta sección tenían como objetivo conocer los problemas de los estudiantes como: la carga académica, la planificación que tienen sobre sus actividades y las razones por las cuales ellos consideran que han dejado de entregar alguna tarea o algún proyecto.

Figura 5. Descripción de la pregunta número 4 y su resultado.



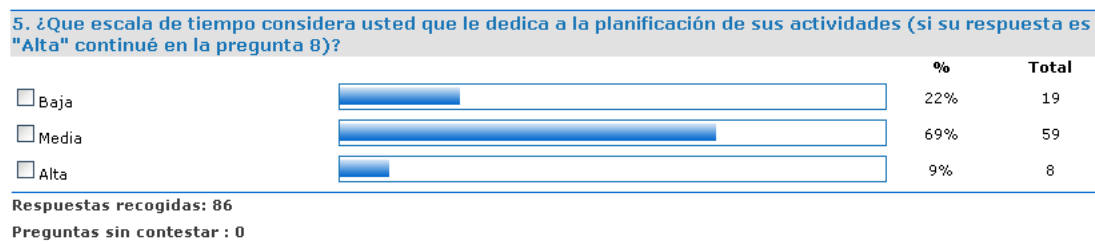
Fuente: Luis A. Pusey, **Proyecto Sak'a**,

<http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183>.

Resultados pregunta 4

La sobrecarga académica es un factor clave para evaluar la factibilidad del proyecto. Podemos observar que el 72% de los estudiantes encuestados sienten que la carga generada por sus carreras es bastante alta.

Figura 6. Descripción de la pregunta número 5 y su resultado.



Fuente: Luis A. Pusey, **Proyecto Sak'a,**

[http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183.](http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183)

Resultados pregunta 5

Se observa que el tiempo dedicado a la planificación de sus actividades es media y pocos dedican gran parte de su tiempo a planificar sus actividades y compromisos con sus estudios.

Figura 7. Descripción de la pregunta número 6 y su resultado.



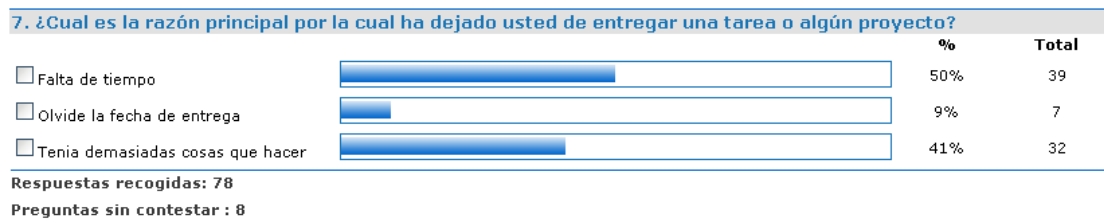
Fuente: Luis A. Pusey, **Proyecto Sak'a,**

[http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183.](http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183)

Resultados pregunta 6

Los estudiantes, en su mayoría, consideran que la falta de tiempo es uno de los factores más grandes para que no planifiquen sus actividades y eventos. Otra causa grande es el no tener conocimiento sobre las herramientas que facilitan esta planificación.

Figura 8. Descripción de la pregunta número 7 y su resultado.



Fuente: Luis A. Pusey, **Proyecto Sak'a,**

<http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183>.

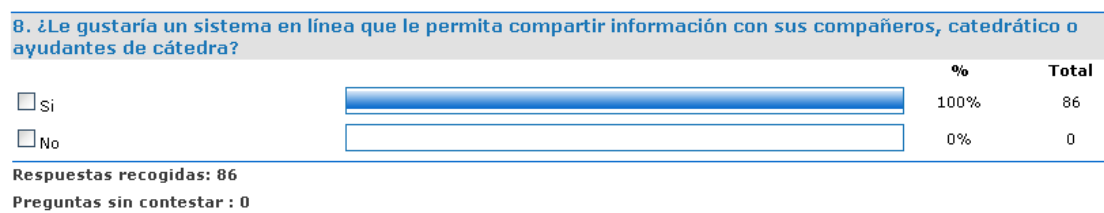
Resultados pregunta 7

Vemos que la mayoría de estudiantes han faltado a tareas o eventos debido a la sobrecarga académica o la mala planificación de los mismos.

Sección 3

Las siguientes preguntas se enfocan en los intereses y las necesidades de los estudiantes universitarios sobre estos sistemas, para afrontar la sobrecarga académica de sus carreras.

Figura 9. Descripción de la pregunta número 8 y su resultado.



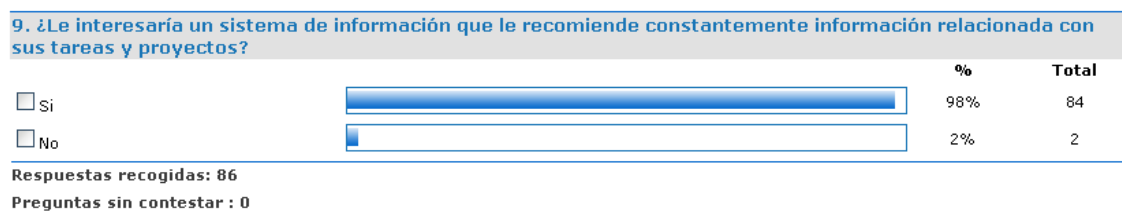
Fuente: Luis A. Pusey, **Proyecto Sak'a,**

<http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183>.

Resultados pregunta 8

El 100% de los estudiantes universitarios consideran necesario y les gustaría poseer una herramienta en la que puedan contactar y compartir información con sus compañeros de estudio y catedráticos.

Figura 10. Descripción de la pregunta número 9 y su resultado.



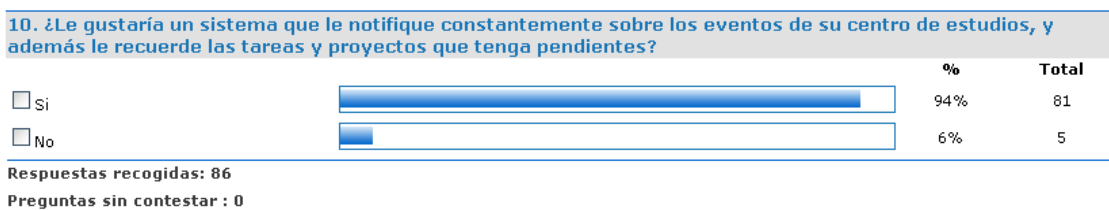
Fuente: Luis A. Pusey, **Proyecto Sak'a,**

<http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183>.

Resultados pregunta 9

El 98% de los estudiantes universitarios está interesado en un sistema de información que le ayude y recomiende constantemente información relacionada con sus tareas, proyectos y asignaciones de su carrera.

Figura 11. Descripción de la pregunta número 10 y su resultado.



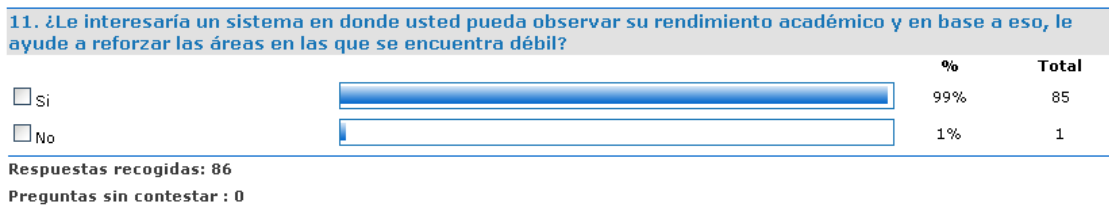
Fuente: Luis A. Pusey, **Proyecto Sak'a,**

<http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183>.

Resultados pregunta 10

El 94% de los estudiantes universitarios está interesado en un sistema de información que le recuerde de sus actividades (tareas, eventos, etc.) de su centro de estudios y demás.

Figura 12. Descripción de la pregunta número 11 y su resultado.



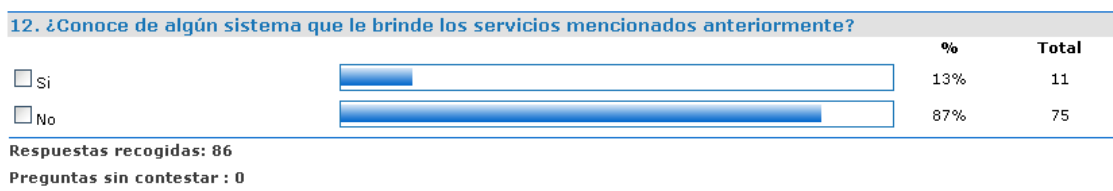
Fuente: Luis A. Pusey, **Proyecto Sak'a,**

<http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183>.

Resultados pregunta 11

El 99% de los estudiantes universitarios está interesado en un sistema de información donde pueda observar su rendimiento académico y en base a eso, le ayude a reforzar las áreas donde se encuentra un poco débil.

Figura 13. Descripción de la pregunta número 12 y su resultado.



Fuente: Luis A. Pusey, **Proyecto Sak'a,**

<http://encuestafacil.com/RespWeb/Qn.aspx?EID=75183>.

Resultados pregunta 12

Como podemos observar, el 87% de los estudiantes universitarios no conocen sistemas de información que realicen todas las actividades anteriormente mencionadas y muchos de los que si conocen, pero no las utilizan por ser sistemas pagados.

3.3.2 Comentarios sobre los resultados de las encuestas

Al analizar la encuesta podemos observar que el 60% de los encuestados, no están satisfechos con el sistema de información que les facilita la universidad. Podemos observar también que debido a la sobrecarga académica de sus carreras, el 91% de los encuestados no toma el debido tiempo para planificar sus actividades y eventos, por lo que es de su interés un sistema que les facilite la planificación de actividades.

Los servicios que ofrecen generalmente los sistemas de información son: comunicación directa con el catedrático, comunicación directa con el ayudante de cátedra y notificación de los eventos importantes del centro de estudios. Es lamentable que en una menor escala se encuentre la comunicación entre los compañeros de clases, debido a que es fundamental para compartir y aprovechar de una mejor forma el conocimiento adquirido, por lo que este sería un requerimiento indispensable para los futuros sistemas de información.

El 99% de los encuestados mostraron interés por un sistema que les recomiende y les permita compartir información relacionada con sus tareas y proyectos. Por otra parte demandan una herramienta fácil de usar que les ayude a organizar su tiempo y que también lleve el control de su rendimiento académico.

4 REQUERIMIENTOS DEL SISTEMA

4.1 Administrar grupos

Permite administrar el mantenimiento de los grupos dentro del sitio con la finalidad de crear un colaboratorio con mismos objetivos e interés, pudiendo así, intercambiar información, conocimiento, noticias de interés, etc.

Actores:

- Administrador de grupos
- Operador de grupos

Precondiciones:

- Parametrización de tipos de usuario
- Parametrización de tipos de grupo

Escenarios

Escenario 1: agregar grupo

Permite la creación de un grupo dentro del sistema, según ciertas áreas de interés.

Escenario 2: actualizar grupo

Permite la modificación de un grupo dentro del sistema, el actor debe tener los suficientes permisos para realizar esta acción.

Escenario 3: eliminar grupo

Permite la eliminación de un grupo dentro del sistema, el actor debe tener los suficientes permisos para realizar esta acción.

Escenario 4: consultar grupo

Permite la consulta de un grupo dentro del sistema, el actor debe tener los suficientes permisos para realizar esta acción.

Escenario 5: asignar grupo

Permite la asignación a un grupo dentro del sistema, el actor debe tener los suficientes permisos para realizar esta acción.

Escenario 6: desasignar grupo

Permite la modificación de un grupo dentro del sistema, el actor debe tener los suficientes permisos para realizar esta acción.

4.2 Administrar horarios

Permite proponer un horario de trabajo sobre un grupo, esto con el fin de tener a varios afiliados a cierta hora en línea, creando así un centro de estudiantes y/o catedráticos que comparten los mismos intereses sobre un tema en un horario.

Actores:

- Operador de horarios
- Administrador de horarios

Precondiciones:

- Parametrización de tipos de usuario
- Parametrización de tipos de grupo

Escenarios

Escenario 1: agregar horario

Permite crear una propuesta de horario dentro de un grupo, esto con el fin de reunir a la mayoría de los afiliados en línea.

Escenario 2: asignarse a horario

Permite asignarse a un horario ya creado dentro del sistema y dentro del grupo. El actor debe tener los suficientes permisos para realizar esta acción.

Escenario 3: desasignarse a horario

Permite des asignarse a un horario al cual ya se estaba registrado dentro del grupo.

Escenario 4: consultar horario

Permite consultar a detalle un horario ya creado dentro del sistema y dentro del grupo.

Escenario 5: eliminar horario

Permite la eliminación de un horario dentro del sistema, el actor debe tener los suficientes permisos para realizar esta acción y que no se haya utilizado el mismo.

4.3 Administrar eventos

El sistema permite la parametrización de eventos, de modo que estos recuerden al alumno las horas en las que debe trabajar sobre las actividades del grupo al que esté registrado mediante alarmas, ya sean textuales o sonoras, según él decida.

Actores:

- Operador de eventos
- Administrador de eventos

Precondiciones:

- Parametrización de tipos de usuario
- Parametrización de tipos de eventos
- Parametrización de tipos de alarmas

Escenarios**Escenario 1: agregar evento**

Permite crear un nuevo evento dentro del sistema. Ingresando las preferencias sobre la notificación y tipo.

Escenario 2: actualizar evento

Permite modificar un evento previamente creado. Esto solo para los eventos que no se hayan realizado.

Escenario 3: eliminar evento

Permite eliminar un evento dentro del sistema que haya creado con anterioridad.

Escenario 4: consultar evento

Permite eliminar un evento dentro del sistema que haya creado con anterioridad.

4.4 Notificar eventos

El sistema, según el horario establecido, recordará al estudiante las horas en las que debe trabajar sobre las actividades del grupo al que esté registrado mediante alarmas, ya sean textuales o sonoras, según sean parametrizadas. Se le notificarán horarios de envío de tareas, mensajes del catedrático, horarios de salida hacia un evento o clase y noticias relevantes al dominio en el que se desenvuelva (nuevos acontecimientos, suspensión de clases, cambio de salones, cambio de horarios, fechas de asignaciones e inscripciones, etc.).

Actores:

- Reloj del sistema
- Precondiciones
- Parametrización de tipos de eventos
- Parametrización de tipos de alarmas

Escenarios

Escenario 1: notificar evento

Notifica un evento creado en el sistema. Verifica el tiempo y las acciones a realizar sobre el mismo.

4.5 Administrar documentos

Este caso de uso permite al usuario compartir información con la comunidad y administrarla, además tiene la capacidad de darle una puntuación y avisar a todos los usuarios dentro del grupo sobre dicha publicación.

Actores:

- Operador de documentos
- Administrador de documentos

Precondiciones:

- Parametrización de tipos de documento
- Parametrización de tipos de usuario
- Parametrización de tipos de alarmas
- Parametrización de tipos de eventos
- Parametrización de estados de documentos

Escenarios

Escenario 1: publicar documento

Permite publicar un documento dentro del grupo en cuestión para ser compartido con la comunidad. Además de parametrizar el tipo de aviso que debe de dar sobre la publicación y la puntuación asignada al mismo.

Escenario 2: eliminar documento

Permite dar de baja un documento dentro del grupo en cuestión, debe poseer permisos para dicha acción. Además debe parametrizar el tipo de aviso a dar sobre dicha eliminación.

Escenario 3: consultar documento

Permite consultar un documento que pertenece al grupo en cuestión, siempre y cuando tenga los permisos asociados para consulta.

Escenario 4: comentar documento

Permite comentar un documento dentro del sistema que pertenece al grupo, siempre y cuando tenga los permisos asociados para dicha acción.

Escenario 5: valorar documento

Permite valorar un documento dentro del sistema que pertenece al grupo, siempre y cuando tenga los permisos asociados para dicha acción.

4.6 Administrar enlaces

Este caso de uso permite al usuario compartir enlaces de interés con la comunidad y administrarlos como parte del grupo, además tiene la capacidad de darle una puntuación y avisar a todos los usuarios dentro del grupo sobre dicha publicación.

Actores:

- Operador de enlaces
- Administrador de enlaces

Precondiciones:

- Parametrización de tipos de enlaces
- Parametrización de tipos de usuario
- Parametrización de tipos de alarmas
- Parametrización de tipos de eventos
- Parametrización de estados de documentos

Escenarios**Escenario 1: publicar enlace**

Permite publicar un enlace dentro del grupo en cuestión para ser compartido con la comunidad. Además de parametrizar el tipo de aviso que debe de dar sobre la publicación y la puntuación asignada al mismo.

Escenario 2: eliminar enlace

Permite dar de baja un enlace dentro del grupo en cuestión, debe poseer permisos para dicha acción. Además debe parametrizar el tipo de aviso a dar sobre dicha eliminación.

Escenario 3: consultar enlace

Permite consultar un enlace que pertenece al grupo en cuestión, siempre y cuando tenga los permisos asociados para consulta.

Escenario 4: comentar enlace

Permite comentar un enlace dentro del sistema que pertenece al grupo, siempre y cuando tenga los permisos asociados para dicha acción.

Escenario 5: valorar enlace

Permite valorar un enlace dentro del sistema que pertenece al grupo, siempre y cuando tenga los permisos asociados para dicha acción.

4.7 Administrar cuestionarios

Permite al actor publicar y administrar los cuestionarios. El objetivo de estos es la publicación para que posteriormente los usuarios puedan reforzar las áreas más débiles que el actor posea sobre un tema.

Actores:

- Operador de cuestionarios
- Administrador de cuestionarios

Precondiciones:

- Parametrización de tipos de usuario
- Parametrización de tipos de alarmas
- Parametrización de tipos de eventos
- Parametrización de tipos de cuestionarios
- Parametrización de tipos de preguntas
- Parametrización de tipos de respuestas
- Parametrización de estados de documentos

Escenarios

Escenario 1: agregar cuestionarios

Permite al actor agregar cuestionarios para evaluar o reforzar temas del grupo de estudio. Se debe tener los permisos adecuados para publicar un cuestionario.

Escenario 2: actualizar cuestionario

Permite al actor actualizar cuestionarios, por algún dato erróneo ingresado o cambiar la Parametrización del mismo. Se debe tener los permisos adecuados para realizar un cambio sobre un cuestionario.

Escenario 3: eliminar cuestionario

Permite dar de baja un cuestionario dentro del grupo, debe poseer permisos para dicha acción. Además debe parametrizar el tipo de aviso a dar sobre dicha eliminación.

Escenario 4: agregar pregunta

Permite agregar una pregunta dentro de un cuestionario que pertenece al grupo, siempre y cuando tenga los permisos asociados para dicha acción.

Escenario 5: actualizar pregunta

Permite actualizar los datos de una pregunta de un cuestionario, siempre y cuando tenga los permisos asociados para dicha acción.

Escenario 6: eliminar pregunta

Permite dar de baja una pregunta de un cuestionario, siempre y cuando tenga los permisos asociados para dicha acción.

4.8 Contestar cuestionario

Permite al actor contestar los cuestionarios. El objetivo de estos es reforzar las áreas más débiles que el actor posea sobre un tema.

Actores:

- Operador de cuestionarios
- Administrador de cuestionarios

Precondiciones:

- Parametrización de tipos de usuario
- Parametrización de tipos de alarmas
- Parametrización de tipos de eventos
- Parametrización de tipos de cuestionarios
- Parametrización de tipos de preguntas
- Parametrización de tipos de respuestas
- Parametrización de estados de documentos

Escenarios

Escenario 1: contestar cuestionario

Permite contestar un cuestionario publicado en el grupo, con el objetivo de mejorar el rendimiento de los estudiantes o de reforzar un tema en específico.

4.9 Consultar estadísticas

Permite al actor observar estadísticas sobre un estudiante o un grupo en general.

Actores:

- Operador de estadísticas
- Administrador de estadísticas

Precondiciones:

- Parametrización de tipos de usuario
- Parametrización de tipos de alarmas
- Parametrización de tipos de eventos
- Parametrización de tipo de estadística

Escenarios

Escenario 1: consultar estadística

Permite al actor observar estadísticas sobre un estudiante o un grupo en general. Se debe tener los permisos adecuados para realizar dicha acción.

4.10 Administrar mensajes

Este caso de uso permite al administrar un sistema parecido al de correo, pudiendo enviar, consultar, eliminar mensajes a cierto grupo de usuarios sobre noticias, eventos, etc.

Actores:

- Administrador de mensajes
- Operador de mensajes

Precondiciones:

- Parametrización de tipos de usuario
- Parametrización de tipos de alarmas
- Parametrización de tipos de eventos
- Parametrización de tipo de mensajes
- Parametrización de prioridades

Escenarios

Escenario 1: enviar mensaje

Permite el envío de mensajes internos dentro de los contactos que se posean. Se debe tener los permisos adecuados para realizar dicha acción.

Escenario 2: consultar mensaje

Permite al actor consultar los mensajes que se le han enviado. Informándoles sobre noticias, eventos, etc.

Escenario 3: eliminar mensaje

Permite al actor eliminar mensajes que tiene en su bandeja de entrada.

Escenario 4: reenviar mensaje

Permite al actor reenviar mensajes que le han llegado, para uno o varios destinatarios.

Escenario 5: responder mensaje

Permite al actor observar estadísticas de su rendimiento y consultar información acerca del grupo. Se debe tener los permisos adecuados para realizar dicha acción.

4.11 Administrar cuenta

Por medio de este caso de uso el actor puede administrar su cuenta en el sitio. Una vez creada la cuenta el actor tendrá privilegios para crear grupos o bien para unirse a uno que ya este creado y para compartir información.

Actores:

- Administrador de cuentas

Precondiciones:

- Parametrización de tipo de usuario
- Parametrización de tipo de grupo

Escenarios

Escenario 1: crear cuenta

Permite crear cuentas dentro del sistema, para poder registrarse a un grupo o asociación y poder beneficiarse de la información de la comunidad.

Escenario 2: editar cuenta

Permite modificar una cuenta previamente creada. Para modificar perfiles.

Escenario 3: cerrar cuenta

Permite cerrar una cuenta creada.

4.12 Administrar *banners*

Por medio de este caso de uso el actor podrá administrar los *banners* ya sea de publicidad o de información del sitio.

Actores:

- Administrador de *banners*
- Precondiciones
- Parametrización de tipo de usuario
- Parametrización de tipo de *banner*

Escenarios

Escenario 1: agregar *banner*

Permite cargar un *banner* dentro del sitio.

Escenario 2: eliminar *banner*

Permite modificar un *banner* dentro del sitio.

4.13 Administrar noticias

Por medio de este caso de uso el actor podrá administrar las noticias a publicar dentro del sitio.

Actores:

- Administrador de noticias
- Precondiciones
- Parametrización de tipo de usuario
- Parametrización de tipo de noticia

Escenarios**Escenario 1: agregar noticia**

Permite cargar una nueva noticia al sitio.

Escenario 2: eliminar noticia

Permite eliminar una noticia previamente cargada.

4.14 Administrar contactos

Permite a los usuarios administrar sus contactos dentro del sistema. Esto con el fin de tener comunicación con los usuarios del grupo y otros usuarios que están registrados en el sistema.

Actores:

- Administrador de contactos

Precondiciones:

- Parametrización de tipos de usuario
- Parametrización de tipos de contacto

Escenarios**Escenario 1: agregar contacto**

Permite agregar un contacto que no haya sido previamente ingresado por la asociación de un grupo.

Escenario 2: actualizar datos de contacto

Permite la modificación de los datos que se ingresaron sobre un contacto, para manejarlo de una mejor manera.

Escenario 3: bloquear contacto

Permite el bloqueo de un contacto dentro del sistema, con esta opción no recibes las notificaciones del mismo.

Escenario 4: consultar contacto

Permite la consulta de un contacto asociado.

4.15 Administrar foro

Permite a los usuarios administrar los temas de los que se pueden comentar ya sea dentro de un grupo o a nivel del sistema. Esto con el fin de tener un centralizar la información sobre un tema en específico.

Actores:

- Administrador de temas
- Operador de temas

Precondiciones:

- Parametrización de tipos de usuarios
- Parametrización de tipos de temas

Escenarios

Escenario 1: agregar tema

Permite agregar un tema para poder comentar sobre el mismo ya sea dentro del grupo o a nivel del sistema.

Escenario 2: actualizar tema

Permite la modificación de los datos que se ingresaron sobre el tema.

Escenario 3: eliminar tema

Permite la eliminación de un tema.

Escenario 4: consultar tema

Permite la consulta de un tema publicado.

Escenario 5: comentar tema

Permite hacer comentarios sobre un tema publicado.

Escenario 6: responder a comentario

Permite responder un comentario previamente ingresado para dejar un registro de a que comentario se está respondiendo.

4.16 Administrar integrantes del grupo

Permite a los usuarios dueños de un grupo administrar los integrantes del mismo. Esto con el fin de poder tener un grupo controlado.

Actores:

- Administrador de grupo.

Precondiciones:

- Parametrización de tipos de usuario.
- Parametrización de tipos de integrantes

Escenarios

Escenario 1: aceptar integrante

Permite aceptar una solicitud de un usuario para ingresar al grupo.

Escenario 2: rechazar integrante

Permite aceptar una solicitud de un usuario para ingresar al grupo.

Escenario 3: editar integrante

Permite editar los permisos y ciertas características de los integrantes del grupo.

Escenario 4: invitar integrante

Permite enviar una invitación a un usuario para asociarse al grupo.

4.17 Buscar información

Realiza las búsquedas de la información evaluando los requerimientos del usuario, para poder brindar soporte sobre una asignación o simplemente informar al usuario de fuentes donde puede profundizar ciertos temas.

Actores:

- Administrador de Información

Precondiciones:

- Parametrización de tipos de usuario
- Parametrización de tipos de grupo

Escenarios**Escenario 1: buscar información por usuario**

Realiza una búsqueda de información que ha visitado o publicada con el usuario.

Escenario 2: buscar información por etiqueta

Realiza una búsqueda de información según es solicitada por el usuario mediante asignaciones nuevas y solicitudes por medio de las etiquetas de cada publicación del sistema.

5 MODELO DE DISEÑO DETALLADO DEL SISTEMA

5.1 Metas y restricciones arquitectónicas

En este punto se van a definir las metas y las restricciones que se deben cumplir en la arquitectura de *software* del sistema:

1. La arquitectura se basará sobre el modelo MVC, por lo cual se ha definido como ambiente de desarrollo de *Hibernate* para el desarrollo de la aplicación.
2. La arquitectura del sistema se enfoca en un servidor de aplicaciones que será accedido por medios *web*, ingresando y creando una sesión en el sistema.
3. El sistema será implementado con la arquitectura *web* siendo el servidor central el Servidor de Aplicaciones.
4. El sistema manejará bases de datos MySQL para el manejo de la información.
5. El sistema manejará una completa seguridad y confidencialidad de los datos, utilizando así conexiones seguras por medio de encriptaciones y certificados digitales con claves públicas.
6. La aplicación será implementada por medio de herramientas capaces de agilizar el desempeño de la aplicación, de forma que los usuarios obtengan la mayor funcionalidad y agilidad en sus operaciones. Utilizando la tecnología *Hibernate*.

5.2 Vista de casos de uso

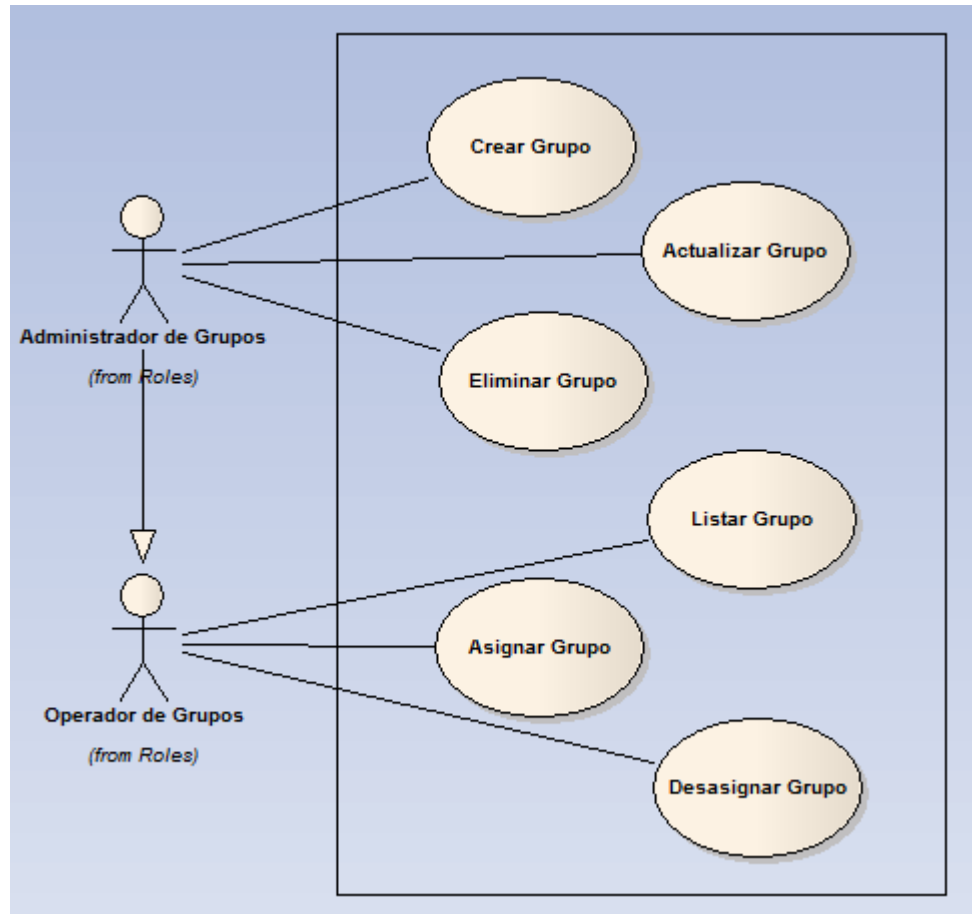
En esta vista se expondrán los casos de uso más críticos del sistema.

5.2.1 Diagramas de casos de uso

Caso de uso: grupos

Permite a los usuarios que posean dichos permisos, administrar el mantenimiento de los grupos dentro del sitio. Con la finalidad de crear un colaboratorio sobre un tema de interés, pudiendo así, intercambiar información, conocimiento, noticias de interés, etc.

Figura 14. Diagrama del caso de uso grupos.

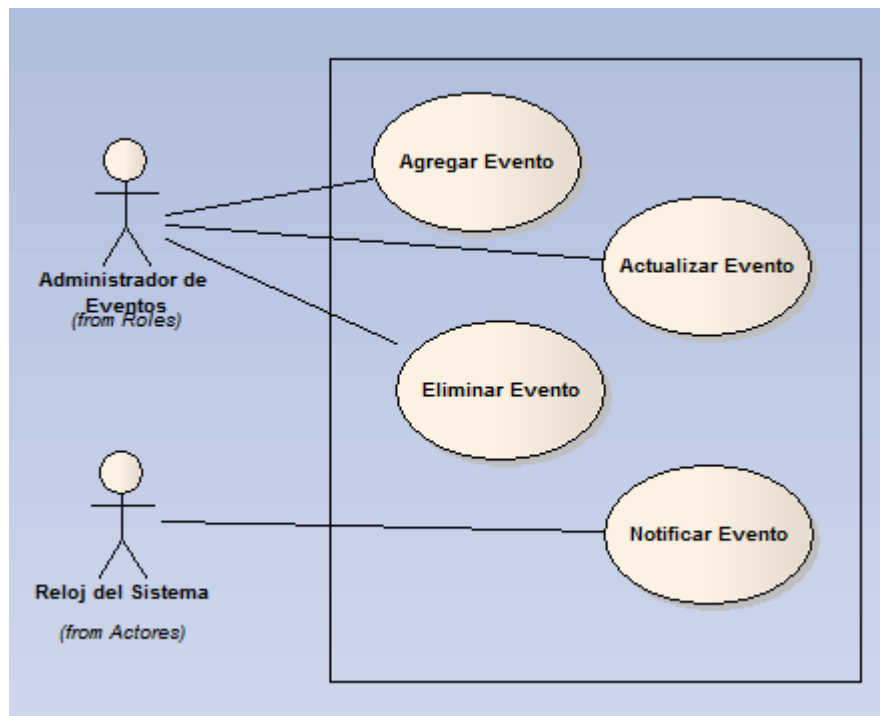


Fuente: Johan Cristales.

Caso de uso: eventos

El sistema, según el horario establecido, recordará al estudiante las horas en las que debe trabajar sobre las actividades del grupo al que esté registrado mediante alarmas, ya sean textuales o sonoras. Se le notificarán horarios de envío de tareas, mensajes del catedrático, horarios de salida hacia un evento o clase y noticias relevantes al dominio en el que se desenvuelva (nuevos acontecimientos, suspensión de clases, cambio de salones, cambio de horarios, fechas de asignaciones e inscripciones, etc.).

Figura 15. Diagrama del caso de uso eventos.

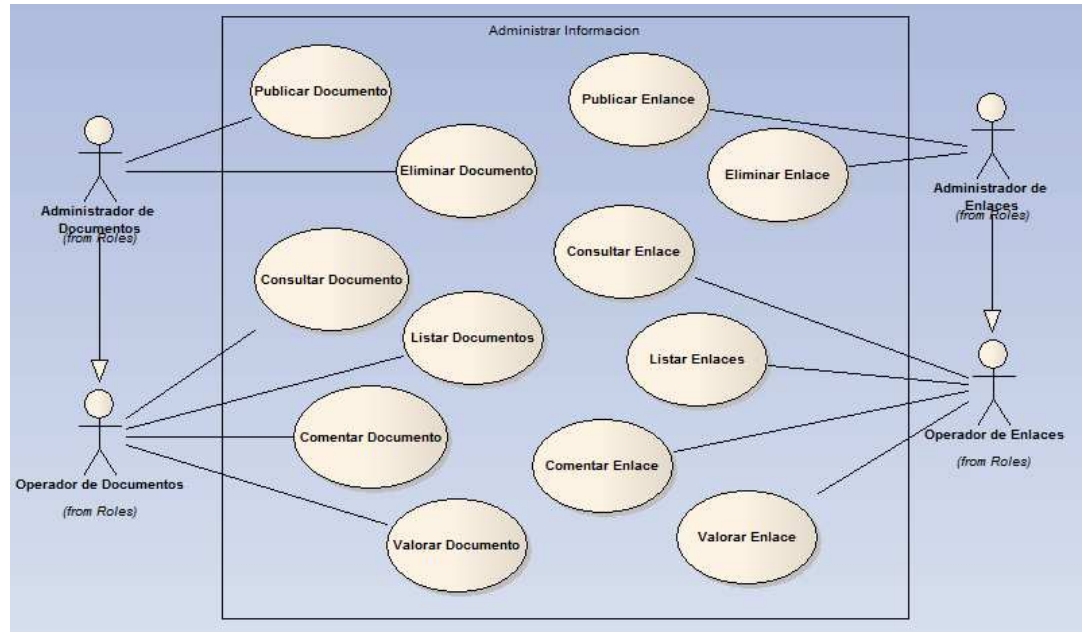


Fuente: Johan Cristales.

Caso de uso: administrar información

Este caso de uso permite al actor compartir información de dos formas con los miembros registrados en un grupo, por medio de documentos y por medio de enlaces, realizando las publicaciones en el servidor para que puedan ser visualizados por las personas registradas al grupo.

Figura 16. Diagrama del caso de uso administrar información.

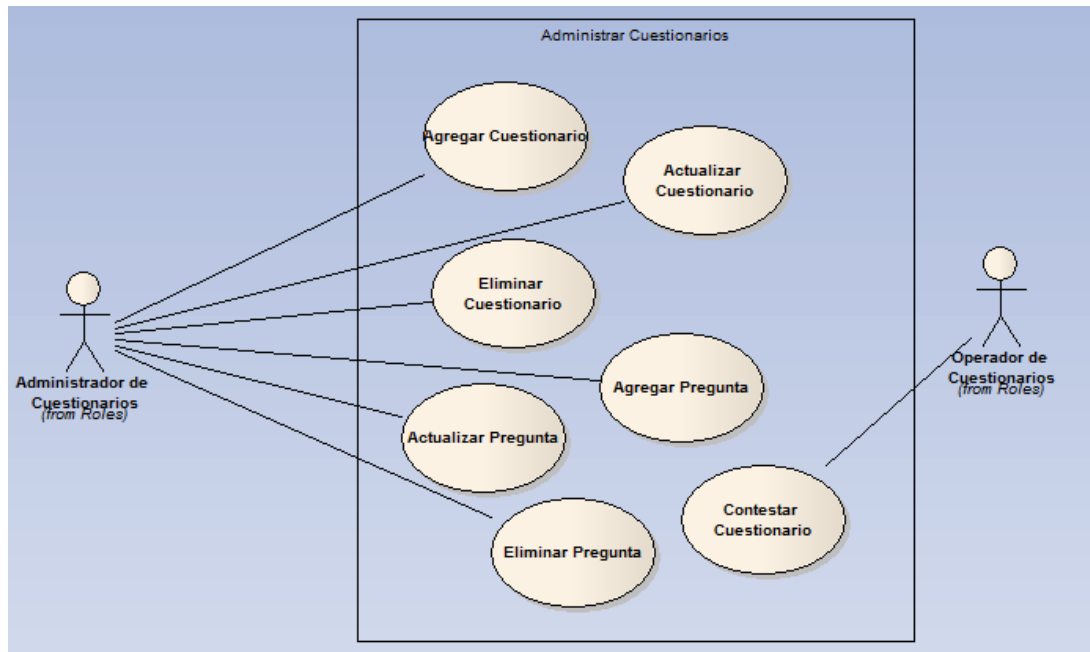


Fuente: Johan Cristales.

Caso de uso: cuestionarios

Este caso de uso permite al actor contestar los cuestionarios ingresados por el dueño del grupo. Los cuestionarios tienen como objetivo reforzar las áreas más débiles que el actor posea sobre un tema.

Figura 17. Diagrama del caso de uso cuestionarios.

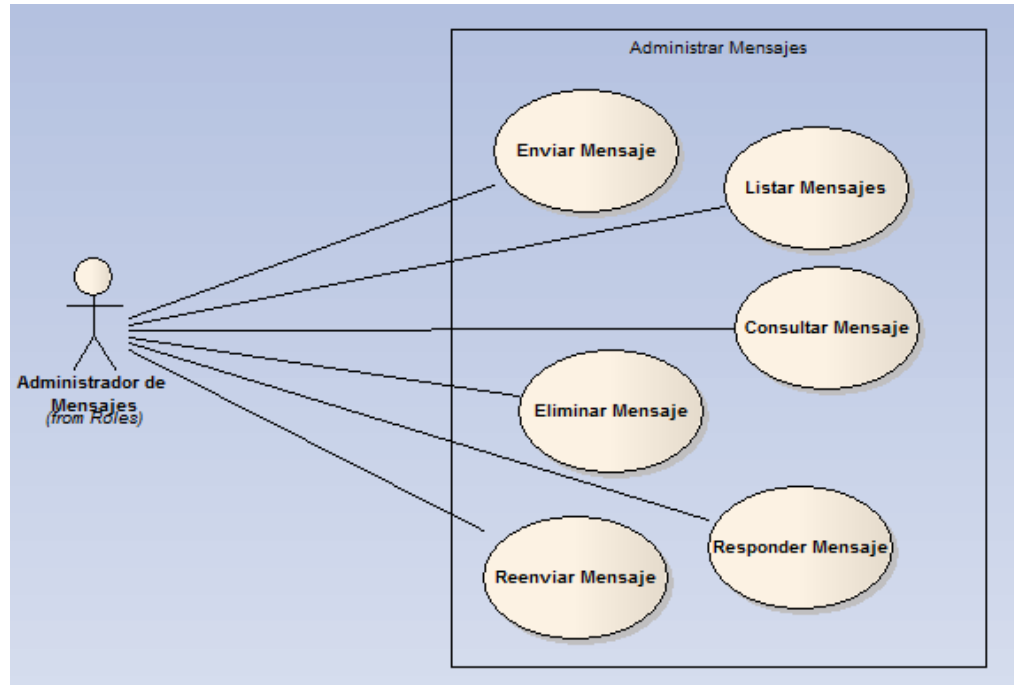


Fuente: Johan Cristales.

Caso de uso: mensajes

Este caso de uso permite al administrar un sistema parecido al de correo, pudiendo enviar, consultar, eliminar, reenviar o responder mensajes a cierto grupo de usuarios sobre noticias, eventos, etc.

Figura 18. Diagrama del caso de uso mensajes.

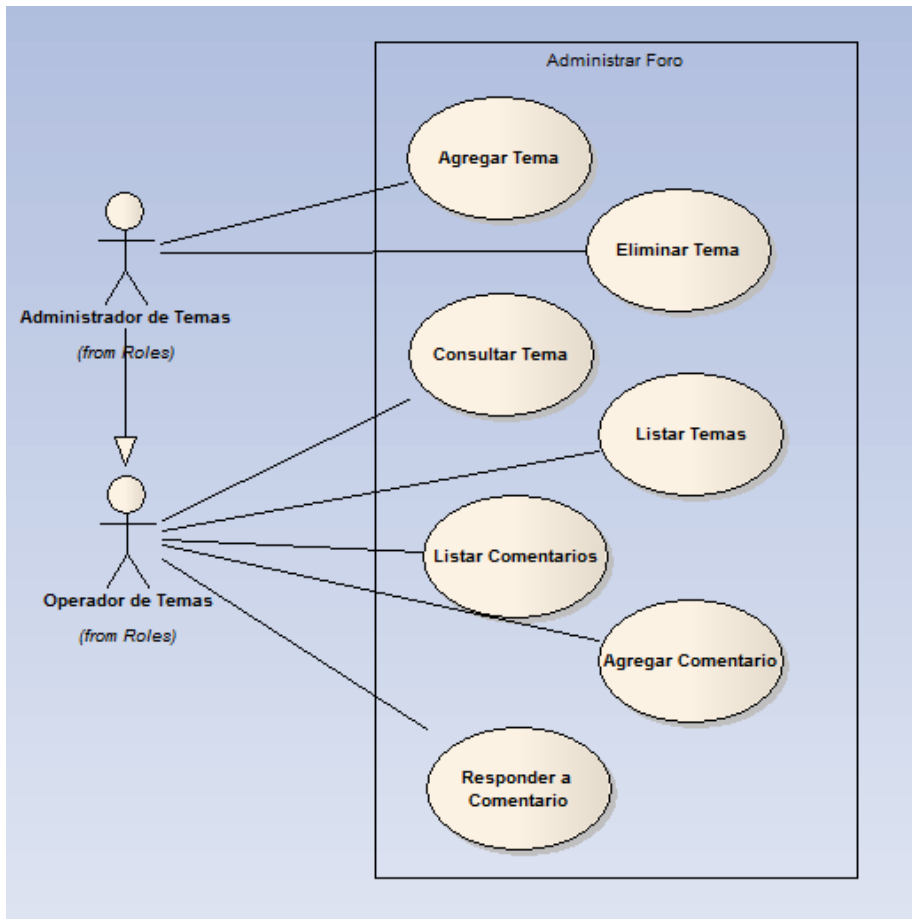


Fuente: Johan Cristales.

Caso de uso: administrar foro

Por medio de este caso de uso el actor podrá administrar un sistema de foros, pudiendo administrar temas y comentarios de los mismos.

Figura 19. Diagrama del caso de uso administrar foro.



Fuente: Johan Cristales.

5.3 Vista lógica

En este punto se va a dar una breve descripción de la vista lógica de la arquitectura de la aplicación. En esta vista se describen las clases más importantes, su organización, los paquetes y subsistemas, y la organización de estos subsistemas dentro de las capas del *software*. Una forma de mostrar las relaciones entre las significantes clases del sistema es por medio de un diagrama de clases, así como también el diagrama de estados.

El objetivo de esta vista es mostrar el sistema desde el punto de vista de los usuarios finales, además se deben de identificar cada uno de los paquetes de la aplicación.

5.3.1 Diagrama de paquetes

El objetivo de este diagrama es mostrar como la aplicación está organizada en paquetes, a continuación se explicará cada unos de estos paquetes con sus respectivas clases que están involucradas:

1. Paquete gt.edu

1.2 *Paquete Saka*: es el paquete que contiene las clases específicas del funcionamiento del sistema, se basa en el modelo de arquitectura MVC.

1.2.1 *Paquete View*: (gt.edu.saka.view), contiene las clases específicas del funcionamiento del sistema que pertenecen al modelo de vista.

1.2.1.1 *Paquete beans*: (gt.edu.saka.view.beans) contiene los beans administrados específicos del funcionamiento del sistema.

1.2.1.2 *Paquete converters*: (gt.edu.saka.converters) contiene los convertidores específicos del funcionamiento del sistema.

1.2.1.3 *Paquete validators*: (gt.edu.saka.validators) contiene los validadores específicos del funcionamiento del sistema.

1.2.2 *Paquete Control*: (gt.edu.saka.control), contiene las clases específicas que pertenecen al modelo de control, estos servicios realizan todo el funcionamiento interno específico del sistema.

1.2.2.1 *Paquete helpers*: (gt.edu.saka.control.helpers): contiene las clases que realizan la funcionalidad de los servicios.

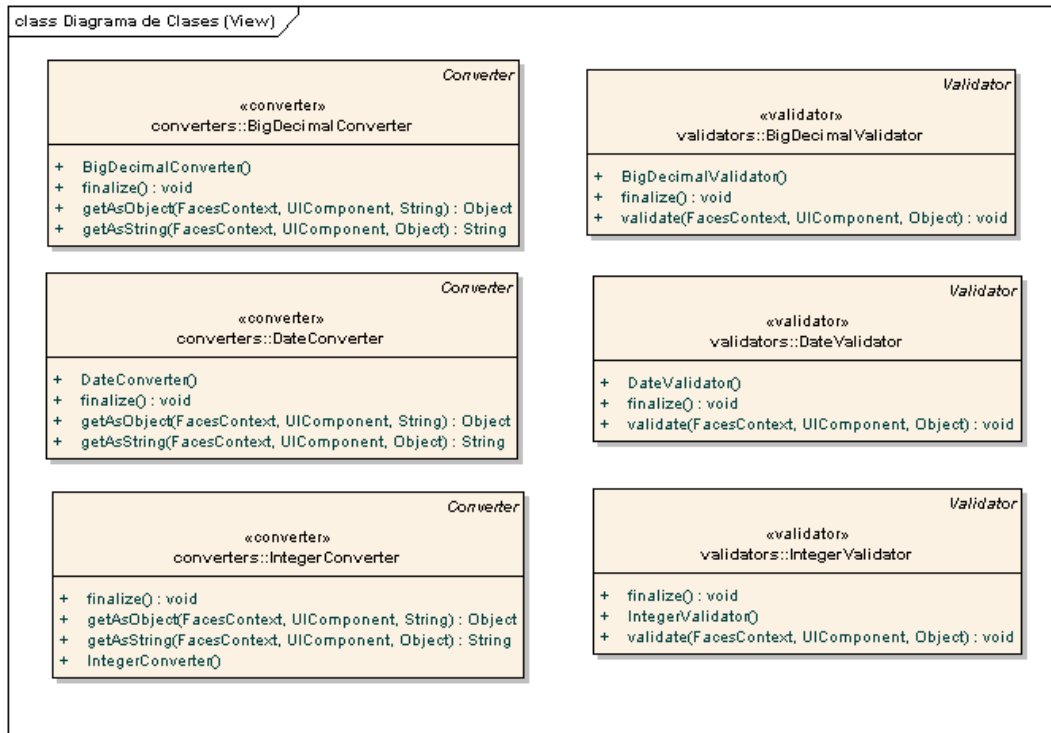
1.2.2.2 *Paquete services*: (gt.edu.saka.control.services): contiene los métodos de toda la funcionalidad del sistema.

1.2.3 *Paquete Model*: (gt.edu.saka.model), contiene las clases específicas que pertenecen al modelo del sistema, estas clases realizan el acceso a la información.

- 1.2.3.1 *Paquete daos:* (gt.edu.saka.model.daos) contiene los métodos que acceden a la base de datos.
- 1.2.3.2 *Paquete information:* (gt.edu.saka.model.information) contiene los objetos de negocio que modelan las tablas de la base de datos
- 1.2.4 *Paquete utils:* (gt.edu.saka.utils), contiene la funcionalidad que es compartida por varios procesos.
- 1.2.5 *Paquete std:* es el paquete que contiene las clases estándar, se basa en el modelo de arquitectura MVC y son básicas para todo el sistema.
 - 1.2.5.1 *Paquete view:* (gt.edu.std.view), Contiene las clases del modelo de vista que son estándar.
 - 1.2.5.2 *Paquete control:* (gt.edu.std.control), contiene los servicios básicos para el sistema.
 - 1.2.5.3 *Paquete model:* (gt.edu.std.model), contiene las clases del modelo básicas para el sistema.
- 1.2.6 *Paquete Webcontent:* contiene todos los elementos *web* del sistema, páginas, imágenes y reportes.
 - 1.2.6.1 *Paquete Saka:* paquete que contiene las páginas y reportes del sistema.
 - 1.2.6.2 *Paquete WEB-INF:* paquete que contiene toda la configuración del sistema en la parte *WEB*.
 - 1.2.6.3 *Paquete std:* paquete que contiene las páginas básicas del sistema.

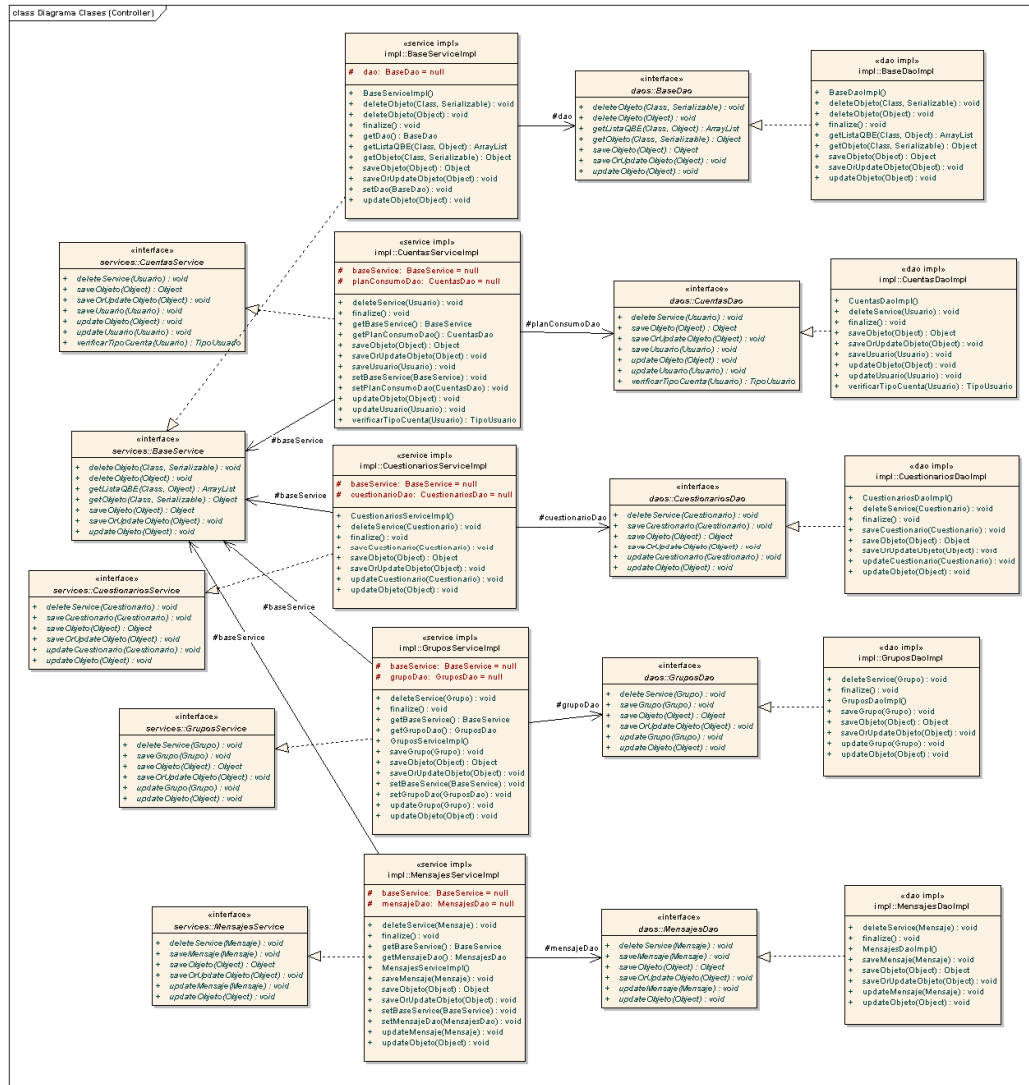
El diagrama de paquetes de cómo el *software* va a estar organizado se visualiza de la siguiente forma:

Figura 21. Diagrama de clases (vista).



Fuente: Cesar García.

Figura 23. Diagrama de clases (controlador).



Fuente: Cesar García.

5.4 Vista física

La arquitectura física del sistema está constituida de la siguiente forma: los usuarios podrán acceder a la aplicación mediante un navegador que será la interfaz entre el usuario y el servidor central. Este servidor estará protegido por un *firewall* que manejará la seguridad de la información personal del usuario como también la información general de la aplicación.

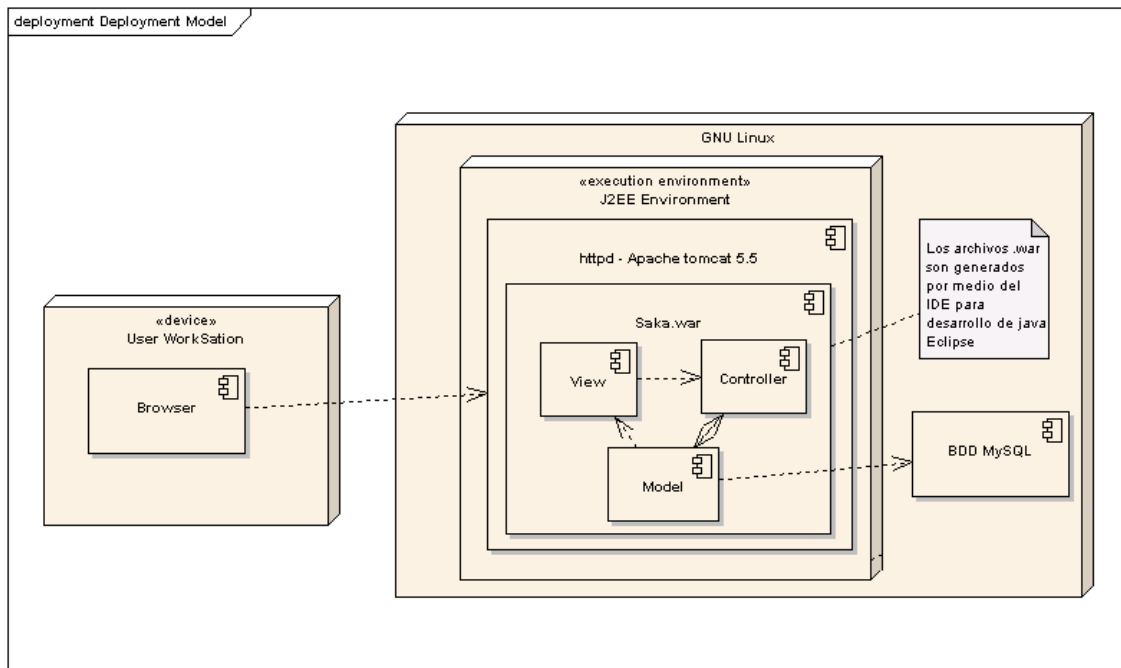
El servidor de aplicaciones contendrá de igual forma con el sitio *web* que podrá ser accedido desde cualquier computadora por medio de *Internet*.

En resumen, esta vista física describe las máquinas donde corre el navegador y que componentes, objetos y procesos son instalados dentro de estas máquinas y sus configuraciones en tiempo de ejecución.

El diagrama *deployment* se visualiza de la siguiente forma:

5.4.1 Diagrama de despliegue

Figura 24. Diagrama de despliegue.

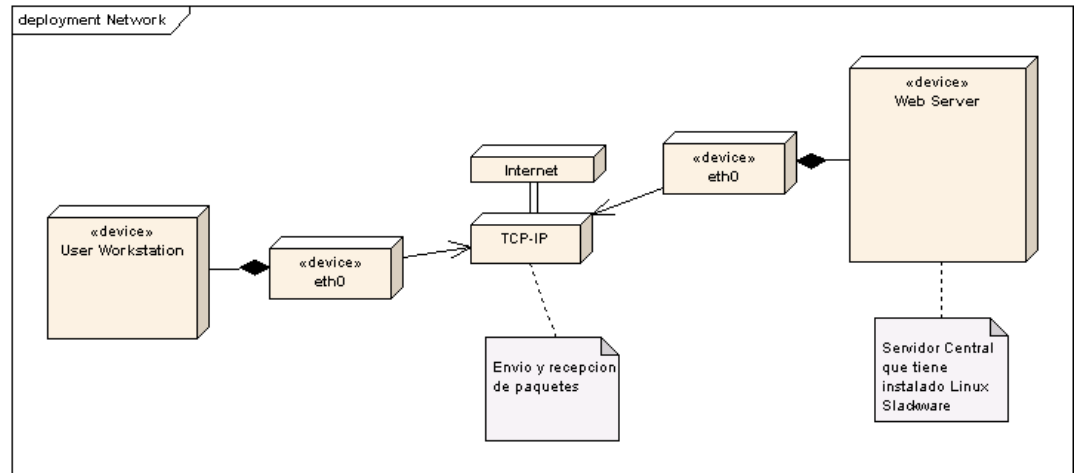


Fuente: Cesar García.

5.4.2 Diagrama de red

El diagrama de red y cada uno de sus componentes los podremos visualizar en la siguiente gráfica:

Figura 25. Diagrama de red.



Fuente: Cesar García.

5.5 Vista de implementación

La vista o arquitectura de desarrollo se enfoca en la organización de módulos del *software*, muestra como las clases están organizadas en paquetes y las dependencias entre los paquetes de *software*. Esta es la vista de diseño que muestra las estructuras de capas del *software* y que responsabilidades tiene cada capa en el sistema.

Las capas del sistema se listan y describen a continuación:

5.5.1 Interfaz usuario – presentación

Esta capa contiene cada una de las clases de los formularios que los actores utilizan para comunicarse con el sistema. El fin de esta capa es proveer las ventanas para la interacción con los usuarios finales de la aplicación. Así como también el formato de los reportes, los menús, las ventanas, tablas, los gráficos y otros elementos importantes en la presentación de los datos al usuario.

Esta capa en el modelo MVC es definida como la vista, esta es la encargada de la interacción con el usuario por medio de componentes UI (Interfaces de usuario) que la referencia de implementación de *Hibernate* provee.

En el modelo JSF esta capa de presentación puede ser acoplada por medio de distintas tecnologías, pero para esta aplicación se utilizará la tecnología JSP. Cada uno de las acciones y valores que corresponden a cada componente UI serán enlazados a *Managed Beans* de Java. Estos serán configurados en el ambiente de *Hibernate* para la comunicación respectiva con los componentes mencionados anteriormente. La respectiva comunicación entre los *Managed Beans* y los componentes UI de los JSP serán enlazados por medio del *Faces Servlet*.

5.5.2 Controlador

Este es la capa que contiene las clases que realizan el trabajo de conexión de la capa mencionada anteriormente y la capa de modelo que se describe a continuación. Existen conceptos importantes a saber en esta capa:

Servicios: clases que contienen métodos que podrán ser accedidos por medio de la capa de presentación para poder obtener o guardar datos. Estos servicios son los que proveen el acceso a los objetos de acceso a los datos (DAOS) que son los que van a traer o guardar datos directamente a la base de datos.

DAO (Objeto de Acceso de Datos): son clases que encapsulan métodos de acceso a datos y que podrán ser accedido por los servicios para devolver o guardar información directamente a la Base de datos.

Service Locator: clase central que será configurada en el ambiente de la aplicación que servirá para localizar cualquier servicio definido en la aplicación, este *Service Locator* podrá ser accedido desde la capa de presentación para poder obtener un servicio definido y este sea el encargado de ir a traer los datos por medio de su respectivo DAO.

El objeto *Service Locator* cumple con el patrón *singleton*, en la que define que únicamente existirá una instancia del objeto *Service Locator*. Este es en si el encargado de enlazar la capa de presentación con la capa de controlador.

BO (Business Object): objetos que serán guardados en la base de datos y que hacen referencia a cada una de las entidades definidas en el modelo de base de datos.

Los DAO's son los encargados de guardar directamente la información a la Base de datos, por medio de JDBC que está definida en la capa de Datos. Este proceso está basado en convertir los BO y guardarlos a la base de datos. O bien obtener registros de la base de datos y convertirlos en sus respectivo BO.

5.5.3 Modelos de negocio

Esta capa da soporte a la capa controlador, el fin de esta capa es proveer la jerarquía de objetos que serán guardados en la base de datos. Esta capa está constituida por las clases y sus respectivas relaciones entre cada una de ellas.

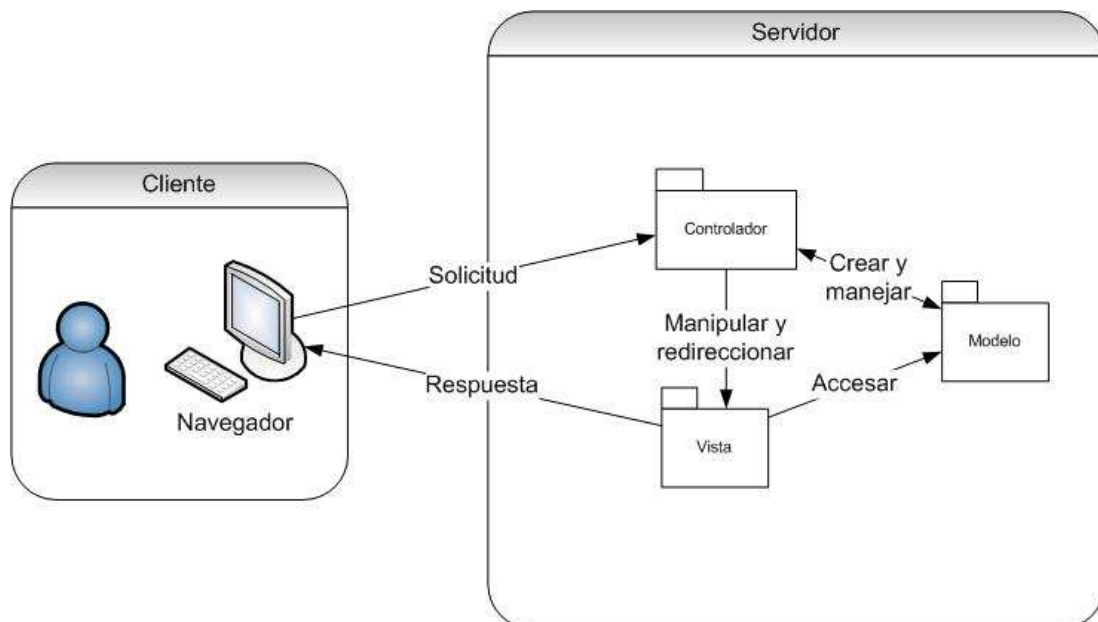
5.5.4 Datos

Esta capa provee el acceso a los datos físicos que la capa controlador está consultando, por ejemplo accesos a la base de datos y conexiones para poder interactuar con dicha base de datos, además esta capa trata acerca de cómo manejar la integridad de los datos de las aplicaciones así como su respectiva seguridad.

Esta capa será manejada por medio de las Clases de Java JDBC localizadas en la plataforma J2EE.

Una forma más clara de poder ver como las capas del *software* se acoplan al modelo a utilizar es por medio del siguiente diagrama:

Figura 26. Diagrama del patrón MVC.



Fuente: Luis Pusey.

5.5.5 Diagrama de componentes

Éste es un diagrama que muestra los componentes identificados con sus respectivas interfaces y relaciones. Se detallan las organizaciones y dependencias lógicas por componente de *software*, ya sea como componentes ejecutables, binarios o código. Éste punto de vista considera todas las decisiones en cuanto al desarrollo, gestión, reutilización y restricciones que define el lenguaje y herramientas de desarrollo tomadas en base a los requerimientos. Estos componentes identificados se convertirán en componentes y paquetes de la infraestructura.

El diagrama de componentes contiene las relaciones, interfaces y reutilización de los mismos, por lo que es una generalización de un diagrama de clases, luego éste se especifica convirtiéndose en diagramas de clases y de casos de uso.

CONCLUSIONES

1. El 80% de los estudiantes encuestados tienen conocimiento del sistema que les provee la universidad para comunicarse con sus catedráticos y compañeros de clase, sin embargo, más del 60% se muestran insatisfechos por el servicio que éste les provee, debido a que nos les facilita la comunicación e interacción con sus compañeros de clase y catedráticos.
2. El 72% de los estudiantes encuestados sienten que la carga generada por sus estudios es alta, adicionalmente a esto, el 69% de los estudiantes no dedican mucho tiempo a la planificación de sus actividades, ya sea por falta de tiempo o porque no conocen herramientas que los apoyen en esta actividad, lo cual genera una desorganización en sus labores como atrasos, desgaste físico y mental, etc.
3. De 100 estudiantes encuestados, un 99% demandan un sistema que les permita: la comunicación, la colaboración y el intercambio de información con sus compañeros de clase y/o catedráticos y la planificación de sus actividades.
4. El *software* desarrollado se basó en la filosofía *Web 2.0*, debido a que facilita a los estudiantes leer y generar conocimiento de forma colectiva, permitiendo que éstos tengan una visión más amplia de los temas de estudio y agilicen el proceso de aprendizaje, además de fomentar la vinculación entre los estudiantes de la carrera.

5. El conocimiento generado por los estudiantes es estructurado por el *software* con una ontología que les permite clasificar y buscar la información con una semántica común entre estudiantes/catedráticos y el *software*.

6. El patrón de arquitectura Modelo Vista Controlador facilitó la separación entre los datos, la lógica de negocio y presentación de la aplicación, adicionalmente facilitó la reutilización de componentes, el mantenimiento y escalabilidad del sistema.

RECOMENDACIONES

1. Informar a los estudiantes sobre los sistemas que provee la universidad para que éstos puedan compartir información e interactuar con sus compañeros de clase y catedráticos.
2. Fomentar en los estudiantes el uso de herramientas *Web 2.0* tales como *wikis*, *blogs*, colaboratorios, etc. Para que estos las puedan utilizar como un medio de apoyo en su aprendizaje y en la planificación de sus actividades.
3. Continuar investigando cómo la *Web 2.0* puede contribuir en la educación y proponer nuevas herramientas que promuevan nuevas formas de aprendizaje.
4. Investigar nuevas tecnologías para desarrollar nuevas versiones del *software* propuesto y cómo este puede implementarse en otras carreras universitarias.
5. Investigar sobre las búsquedas en la ontología por medio de axiomas, *wikis* y agentes inteligentes e implementarlos en nuevas versiones del *software* propuesto.

BIBLIOGRAFÍA

1. Cristóbal Cobo Romani; Hugo Pardo Kuklinski. **Planeta Web 2.0. Inteligencia colectiva o medios fast food.** (Grup de Recerca d'Interaccions Digitals, Universitat de Vic. Flacso México, Barcelona / México DF, 2007).
2. Educastur. **Web 2.0 y Educación.**
http://blog.educastur.es/files/2007/06/web2_0v02.pdf.
3. María Teresa Abad. **Ontologías.**
<http://www.lsi.upc.es/~bejar/ia/material/teoria/3-RC2-Ontologias-2.pdf>.
4. Panamacom. **Glosario de Informática e Internet.**
<http://glosario.panamacom.com/>
5. Flores Macario, Huber Raúl. **Ubicuidad del Aprendizaje Colaborativo.** Trabajo de graduación Ing. en Ciencias y Sistemas, Universidad de San Carlos de Guatemala, Facultad de Ingeniería, 2008.
6. Javier Gramajo López. <http://jgramajo.wordpress.com/>.
7. Chang Ovando, David Armando. **Análisis e Implementación de una Ontología de Actividades Económicas y Productos para PYMES de Guatemala.** Trabajo de graduación Ing. en Ciencias y Sistemas, Universidad de San Carlos de Guatemala, Facultad de Ingeniería, 2007.

