

*Universidad de las Ciencias Informáticas
Facultad 9*



*TÍTULO: CEVALE - Centro Virtual de Autoaprendizaje
de Lenguas Extranjeras. Módulo Recursos.*

*Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas*

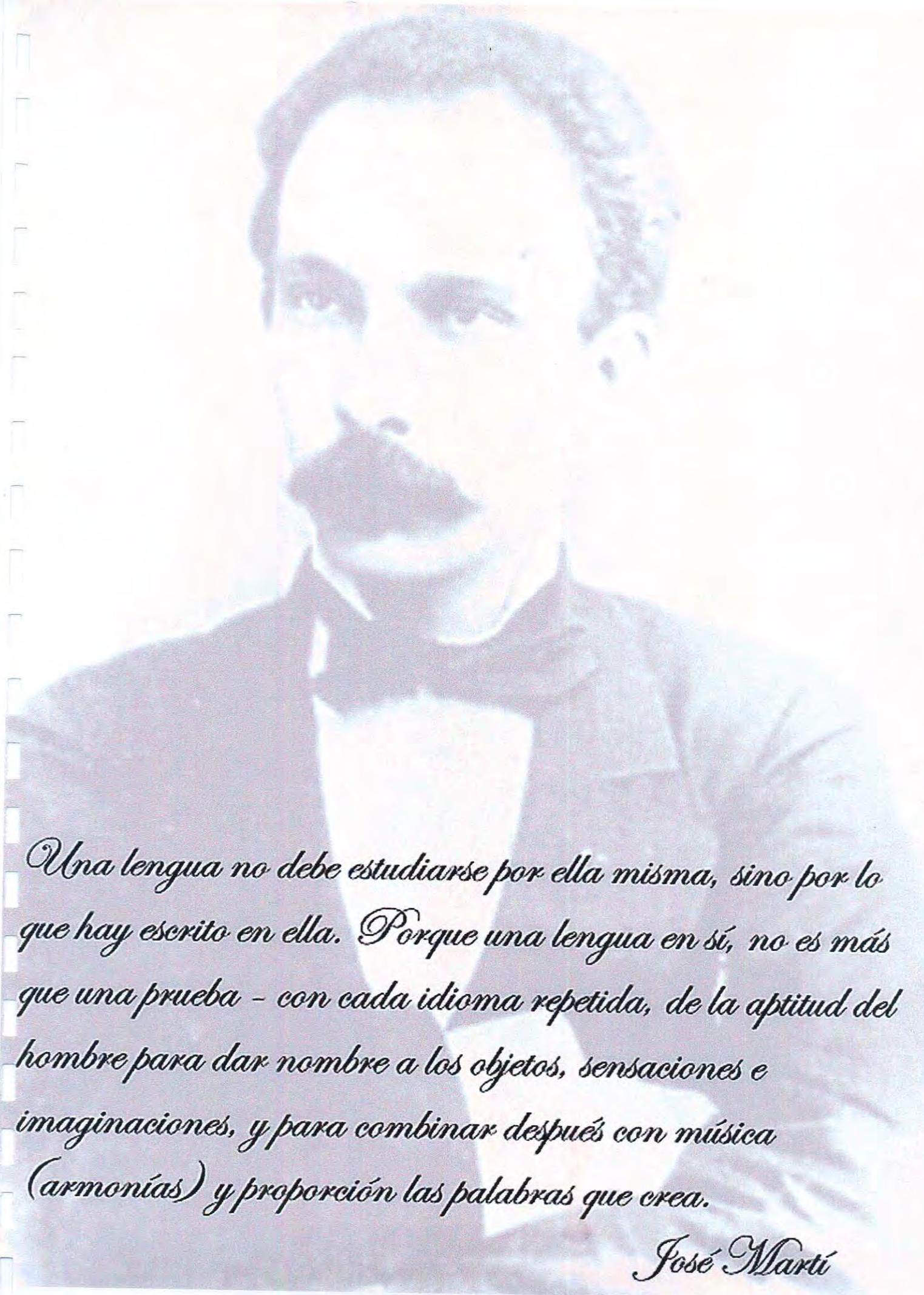
AUTOR: Elizabeth Benítez Ilisástigui

TUTOR: Ing. Jorge Jesús Darias Martínez

CONSULTOR: M.Sc., Arlenys Palmero

Ciudad de La Habana, Cuba Julio de 2008

Año 50 de la Revolución



Una lengua no debe estudiarse por ella misma, sino por lo que hay escrito en ella. Porque una lengua en sí, no es más que una prueba - con cada idioma repetida, de la aptitud del hombre para dar nombre a los objetos, sensaciones e imaginaciones, y para combinar después con música (armonías) y proporción las palabras que crea.

José Martí

DEDICATORIA

A mis padres que me han apoyado en todo momento a lo largo de toda mi vida, ellos que lo han dado todo porque este sueño se haya hecho realidad aguantando todas mis malcriadeces.

A mi abuela con todo mi amor por haber dedicado toda su vida a nosotros, por confiar plenamente en mí, un gran beso.

A Hayro que me daba fuerzas y esperanzas cada día.

AGRADECIMIENTOS

La amistad y el cariño que se recibe de los que te rodean son factores que determinan de forma muy decisiva el desarrollo y equilibrio de una persona. La cualidad más hermosa que debe revestir un ser humano es reconocer a quienes lo han llevado a conseguir sus objetivos.

De forma breve, y tratando de que no me falte nadie por mencionar, quisiera dar mis más grandes agradecimientos a todo aquel que de una forma u otra colaboraron o me apoyaron en la realización de este trabajo.

Mis mayores agradecimientos a mi familia y muy en especial a mi tata Rebeca ,que siempre me dio fuerzas para seguir adelante, a mis padres Norma y Jorge que no dudaron nunca que el día de hoy llegaría y siempre estuvieron presente con su amor y cariño.

Mis amigos que con una sonrisa aliviaban un día en el que no veía que el trabajo adelantaba, en especial Lilliam Menéndez Campo, Yalena Velázquez, Yanisley Mena, Glemnys, Minerva, Ivette, Armando Forrellat, Sergio y muchos otros.

A todos los profesores, por trasmitirnos los conocimientos y valores que nos permitieron formarnos como profesionales, en particular a los profesores Yareisis Pérez Heredia, Elianis, Febe Ciudad Ricardo por sus atenciones y por tendernos la mano en momentos de urgencia.

Al tutor Jorge Darias, por atenderme en todo momento y poner a mi disposición todos sus conocimientos.

A Hayro que siempre estuvo a mi lado y fue la personita que más fuerzas y empeño me dio cada día para culminar este trabajo con éxito.

A todos ustedes, MUCHAS GRACIAS.

DECLARACIÓN DE AUTORÍA

Declaro que Elizabeth Benítez Ilisástigui es la única autora de este trabajo y autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los 5 días del mes de Julio del año 2008.

Elizabeth Benítez Ilisástigui

EBI

Jorge Jesús Darías Martínez

JJM

OPINIÓN DEL TUTOR

El tutor del presente Trabajo de Diploma considera que durante su ejecución, la estudiante Elizabeth Benítez Ilisástigui ha demostrado responsabilidad en la realización de su trabajo de diploma. Ha cumplido con los objetivos trazados en la investigación. La estudiante ha sido independiente, original y autónoma en cada fase de la tarea, logrando resultados satisfactorios. Ha trabajado de manera organizada y planificada. El documento realizado goza de una buena organización, está basado en métodos investigativos científicamente aprobados. Por todo lo anteriormente expresado, considero que la estudiante está apta para ejercer como Ingeniero en Ciencias Informáticas; y propongo que se le otorgue al Trabajo de Diploma la calificación de 5.

A handwritten signature in black ink, appearing to be 'D. Benítez', written in a cursive style.

RESUMEN

La investigación se realiza en los Centros de Autoaprendizaje de las Lenguas Extranjeras (CASIE) en la UCI. La idea de crear el proyecto CEVALE (Centro Virtual de Autoaprendizaje de Lenguas Extranjeras) para la gestión del autoaprendizaje de lenguas extranjeras en la UCI surge por la necesidades existentes en los CASIE expresadas por el Departamento Central de Idiomas Extranjeros (DDC Idiomas) en la UCI para llevar a cabo el proceso de autoaprendizaje. Debido a las desventajas señaladas por los clientes surge la idea de modelar y diseñar una aplicación Web, en este caso, para uno de sus módulos: Módulo Recursos, complaciendo las exigencias y sugerencias de los clientes y usuarios del sistema, además, cuenta con un entorno sencillo y amigable, fácil de instalar y utilizar.

En este trabajo se presenta el análisis y diseño del negocio y el sistema propuesto para dar una solución factible al problema existente utilizando como metodología de construcción de software, El Proceso Unificado de Desarrollo (RUP) y se hace un estudio de los sistemas existentes en Cuba y el mundo y de las herramientas adecuadas para la construcción del sistema.)

PALABRAS CLAVES

- CEVALE
- Aplicación Web
- Modelado
- Gestión de autoaprendizaje

TABLAS Y FIGURAS

Figura 1: Metodología XP	26
Figura 2: Vista general de RUP.....	28
Figura 3: El vocabulario de UML(17).....	31
Figura 4: Diagrama de Casos de Uso del Negocio.....	50
Figura 5: Diagrama de actividades del Caso de Uso Seguir ruta de aprendizaje	60
Figura 6: Diagrama de actividades del Casos de Uso Solicitar Recurso	61
Figura 7: Diagrama de Modelo de Objetos del Negocio	62
Figura 8: Diagrama de Casos de Uso del Sistema.....	66
Figura 9: Arquitectura en 3capas	77
Figura 10: Diagrama de Clases del Análisis CU Gestionar clasificación	81
Figura 11: Diagrama de Clases del Análisis CU Gestionar listado de recursos	82
Figura 12: Diagrama de Clases del Análisis CU Gestionar recursos	82
Figura 13: Diagrama de diseño del CU Gestionar listado de recurso	83
Figura 14: Diagrama de diseño del CU Gestionar recurso	84
Figura 15: Diagrama de diseño del CU Gestionar clasificación	85
Figura 16: Diagrama de Modelo de Despliegue	86
Figura 17: Diagrama de Componentes del CU_Gestionar_Clasificación.....	87
Figura 18: Diagrama de Componentes del CU_Gestionar_List_Recurso	88
Figura 19: Diagrama de Componentes del CU_Gestionar_Recurso	88
Figura 20: Diagrama de Colaboración del CU Gestionar clasificación. Adicionar clasificación	97
Figura 21: Diagrama de Colaboración del CU Gestionar clasificación. Eliminar clasificación	97
Figura 22: Diagrama de Colaboración del CU Gestionar clasificación. Modificar clasificación	98
Figura 23: Diagrama de Colaboración del CU Gestionar recurso. Adicionar recurso	98
Figura 24: Diagrama de Colaboración del CU Gestionar recurso. Eliminar recurso	99
Figura 25: Diagrama de Colaboración del CU Gestionar recurso. Modificar recurso	99
Figura 26: Diagrama de Colaboración del CU Gestionar listado de recursos. Crear lista de recursos	100
Figura 27: Diagrama de actividades del Caso de Uso Atender Usuario	101
Figura 28: Diagrama de actividades del Caso de Uso Localizar Recurso.....	102
Figura 29: Diagrama de actividades del Casos de Uso Realizar encuesta	103
Figura 30: Diagrama de actividades del Casos de Uso Instalar Recurso	104
Figura 31: Diagrama de actividades del Caso de Uso Solicitar Asesoría	105
Tabla 1: Justificación de los actores del negocio.....	48
Tabla 2: Justificación de los trabajadores del negocio	49
Tabla 3: Descripción Textual del Casos de Uso Atender Usuario	52
Tabla 4: Descripción Textual del Casos de Uso Realizar encuesta.....	53
Tabla 5: Descripción Textual del Casos de Uso Solicitar Asesoría.....	54
Tabla 6: Descripción Textual del Casos de Uso Seguir ruta de aprendizaje.....	55
Tabla 7: Descripción Textual del Casos de Uso Solicitar Recurso	56
Tabla 8: Descripción Textual del Casos de Uso Localizar Recurso.....	58
Tabla 9: Descripción Textual del Casos de Uso Instalar Recurso	59
Tabla 10: Justificación de los actores del sistema.....	66
Tabla 11: Descripción Textual del Casos de Uso Gestionar Recurso.....	70
Tabla 12: Descripción Textual del Casos de Uso Gestionar clasificación.....	72
Tabla 13: Descripción Textual del Casos de Uso Gestionar listado de recursos	74

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1: Fundamentación Teórica.....	8
1.1 Introducción.....	8
1.2 Recursos.....	8
1.3 Gestionar recursos sobre un sistema web.....	12
1.4 Sistema Web de Apoyo a la Docencia (SWAD).....	13
1.5 Informatización de los procesos de gestión de los recursos dentro de los CASIE de la UCI.....	15
1.5.1 Centros virtuales de recursos en la UCI.....	15
1.5.2 Sistemas para la gestión de autoaprendizaje vinculados al campo de acción.....	16
1.6 Descripción actual del dominio del problema.....	17
1.7 Situación problemática.....	17
1.8 Propuesta de solución.....	18
1.9 Análisis de otras soluciones existentes.....	19
1.10 Conclusiones parciales.....	23
CAPÍTULO 2: Tendencias y tecnologías actuales a desarrollar.....	24
2.1 Introducción.....	24
2.2 Metodología de desarrollo de software.....	24
2.2.1 Metodología a utilizar (RUP).....	26
2.3 Lenguaje para el Modelado de Objetos.....	29
2.3.1 Lenguaje Unificado de Desarrollo.....	30
2.4 Herramientas a utilizar.....	31
2.4.1 Herramienta de Modelado. Rational Rose.....	31
2.4.2 Herramientas de desarrollo.....	33
2.4.2.1 Lenguajes de Programación.....	33
2.4.2.2 PHP5.....	34
2.4.2.3 PHP vs ASP:.....	36
2.4.3 IDE. Eclipse.....	36
2.5.1 MySQL.....	39
2.6 Conclusiones parciales.....	47
Capítulo 3: Presentación de la solución propuesta.....	48
3.1 Introducción.....	48
3.2 Modelo de Negocio.....	48
3.2.1 Actores y trabajadores del negocio.....	48
3.2.2 Procesos de negocio.....	49
3.2.3 Diagrama de Casos de Uso del Negocio.....	50
3.3 Requerimientos Funcionales.....	63
3.4 Requerimientos No Funcionales.....	63
3.5 Descripción del Sistema Propuesto.....	66
3.5.1 Descripción de los actores del sistema.....	66
3.5.2 Casos de Uso del Sistema.....	66
3.6 Conclusiones parciales.....	74
Capítulo 4: Construcción de la solución propuesta.....	75
4.1 Introducción.....	75
4.2 Arquitectura.....	75
4.2.1 Arquitectura en tres capas.....	76
4.2.2 Patrones.....	78
4.3 Análisis.....	80

4.3.1 Diagramas de Clases del análisis.....	80
4.3.1.1 Diagrama de Clases del Análisis CU Gestionar clasificación.....	81
4.3.1.2 Diagrama de Clases del Análisis CU Gestionar listado de recursos.....	82
4.3.1.3 Diagrama de Clases del Análisis CU Gestionar recursos.....	82
4.4 Diseño.....	83
4.4.1 Diagrama de diseño del CU Gestionar listado de recurso.....	83
4.4.2 Diagrama de diseño del CU Gestionar recurso.....	84
4.4.3 Diagrama de diseño del CU Gestionar clasificación.....	85
4.5 Modelo de Despliegue.....	86
4.6 Modelo de Implementación.....	87
4.7 Conclusiones parciales.....	89
CONCLUSIONES.....	90
RECOMENDACIONES.....	91
REFERENCIAS BIBLIOGRÁFICAS.....	92
ANEXOS.....	97
GLOSARIO DE TÉRMINOS.....	108

INTRODUCCIÓN

Existe en la actualidad un amplio consenso acerca del impacto que tienen las Tecnologías de la Información y las Comunicaciones (TICs) en la sociedad. Las TICs emergen como instrumentos novedosos, generándose una amplia gama de utilidades para éstas, siendo una de las principales, funcionar como eficaces herramientas para el desarrollo. Esta correlación entre TICs y desarrollo ha sido ampliamente estudiada e investigada y, por lo general, se considera que existe una asociación positiva entre ambas variables, de modo que la inversión en TICs se considera una dimensión importante en la consecución exitosa de proyectos de desarrollo. A pesar de esto, falta todavía un largo camino que recorrer para incluir genuinamente a dichas tecnologías en las agendas de desarrollo, dado que los países del Sur no suelen brindar prioridad a la inversión en estas áreas.

La interacción virtuosa TICs-desarrollo es conceptualizada actualmente, a partir de las categorías contenidas en tales objetivos. Así abundan, por ejemplo, investigaciones acerca de la contribución de las TICs a los procesos de desarrollo local mediante el consumo y la producción de TICs (ejemplos de ello son la creación de fuentes de trabajo, la mejora en la calidad de los servicios públicos, el aumento en la productividad del sector privado, etc.(1)

Hace más de cien años el Héroe Nacional cubano, José Martí, afirmó categóricamente: "Ser culto es el único modo de ser libre". Dicha máxima, que mantiene plena vigencia, ha servido a toda la obra de la Revolución Cubana relacionada con el desarrollo socio cultural. Enmarcando cada situación en su momento histórico, hoy en día no se puede hablar de educación, de cultura, de desarrollo, sino se aplican eficientemente y en todas las esferas de la sociedad las nuevas tecnologías de la información y las comunicaciones. La informatización de la sociedad en Cuba forma parte del proyecto revolucionario de elevar la calidad de vida del pueblo cubano y lograr una sociedad cada vez más instruida, educada éticamente, justa, equitativa y solidaria.

Cuba, con un proyecto de desarrollo que tiene como pilares la justicia social, la participación popular, la equidad y la solidaridad, ha diseñado e iniciado la aplicación de estrategias que permiten convertir los conocimientos y las tecnologías de la información y las comunicaciones en instrumentos a disposición del avance y las profundas transformaciones revolucionarias. Numerosos son los logros que se han alcanzado hasta el momento como resultado de los grandes esfuerzos realizados por el Gobierno

Cubano, al punto que se puede decir que las TICs se han insertado en casi todas las ramas de nuestra sociedad.(2)

Actualmente se sigue perfeccionando el trabajo y ampliando el radio de acción de las nuevas tecnologías en beneficio de todas las personas. Se planifican metas ambiciosas que están a la altura de los países del primer mundo y que ya hoy no se está muy lejos de poder alcanzar, ejemplo de ello es la Industria Cubana del Software, con la participación de la Universidad de las Ciencias Informáticas (UCI), y otras empresas productoras de software del país.

Pero estos planes de desarrollo y la puesta en práctica de las tecnologías de la información y las comunicaciones en Cuba se ven seriamente obstaculizados por la política obsesiva, y agresiva del Gobierno de los Estados Unidos, representada por el más férreo bloqueo económico, político y financiero; que además de negar el acceso y disfrute de estas tecnologías las utiliza con fines de subversión política e ideológica.

Cuba ha denunciado en varias ocasiones: foros internacionales, y en cada momento que le es preciso, la política genocida de bloqueo y sus consecuencias para el pueblo cubano y el desarrollo del país; junto a esto, las agresiones de todo tipo por parte del gobierno de Estados Unidos han enlutado y llenado de dolor a múltiples familias cubanas dejando miles de víctimas. A pesar de todo lo anterior el pueblo cubano proseguirá su desarrollo y bajo la dirección de su Gobierno y Partido proseguirá su política de desarrollo en todas las esferas de la sociedad, y en particular a gran escala en la relacionada con las nuevas tecnologías informáticas y las comunicaciones pese a sus modestos recursos materiales y financieros, apoyándose en el conocimiento que ha sabido cultivar en su pueblo. Ejemplo de esto se puede asegurar que en los últimos años, la Informática viene creciendo en los segmentos del quehacer económico y social del país como por ejemplo: la Red Telemática de la Salud, INFOMED, cuenta con servicios de Universidad Virtual, Biblioteca Virtual, Red de Telemedicina, Acceso a Bases de Datos Especializadas y correo electrónico.

No se puede dejar de mencionar el papel protagónico de los Joven Club de Computación y Electrónica, aquí se han capacitado a trabajadores, discapacitados, amas de casa, jóvenes con trastornos de conducta, exreclusos y ancianos. Hoy existen 301 instalaciones, ubicadas en los 169 municipios del país, y cuatro laboratorios móviles para llevar los conocimientos asociados a las tecnologías de la información a zonas de difícil acceso.

Para la Educación Superior Cubana, enfrascada en un proceso de Universalización, ofrecer alternativas de educación de pregrado y postgrado a la totalidad de los ciudadanos del país, en medio de las limitaciones de recursos económicos, constituye un reto sin precedentes, que enfrenta, tomando en cuenta las diferentes modalidades de cursos y los avances significativos alcanzados por las TIC.

En la compleja tarea de hacer productivo el conocimiento, la educación juega un rol determinante, comprometida con el futuro, de contribuir de manera decisiva al conocimiento científico y tecnológico, para lo cual las TIC les ofrecen un soporte de infraestructuras para dar respuesta a las siguientes exigencias sociales: Proporcionar alfabetismo universal de alto nivel, el cual se extiende más allá de los fundamentos de lectura, escritura y aritmética, para abarcar la comprensión básica de las ciencias y de la dinámica de la tecnología, conocimiento de las lenguas extranjeras, aprender a ser eficiente como miembro de una organización, la creación de un pensamiento científico y la alfabetización informática. Infundir en los estudiantes motivación para aprender y disciplina para continuar aprendiendo, es decir, desarrollar la capacidad de aprender a aprender. Estar abierta a las personas de diferentes edades, sexos y contextos socio culturales. Debe ser un sistema abierto que acoja a jóvenes y asegure al mismo tiempo la educación durante toda la vida. La educación no será monopolio de las escuelas, pues será sólo una de las instituciones de aprendizaje, que competirá con los distintos enfoques de instrucción. El aprendizaje organizacional se convierte en una ventaja competitiva sostenible.

La Universidad de Ciencias Informáticas para el desarrollo de la formación académica, investigativa de profesionales en el área de Informática Aplicada se ha sustentado en el desarrollo e integración de las TIC en las soluciones prácticas a diversos problemas sociales desde la educación, lo que ha exigido nuevas interpretaciones de términos como: presencialidad, la comunicación, la interacción, aprendizaje, los materiales educativos, etc. Las condiciones determinantes en el uso de las TIC en la educación con fines educacionales han sido: la proyección gubernamental e institucional, la infraestructura tecnológica y la motivación, preparación de los estudiantes y profesores, aspectos que constituyen el centro de los debates actuales sobre la aplicabilidad y resultados de los programas educativos en plataformas digitales.

Las principales experiencias han estado en la educación presencial sobre plataformas informáticas, desde los modelos de tecnología interactiva consistente en la importancia del interfaz entre el usuario y el sistema y de tecnología colaborativa, relacionada con el grupo y apoyados en la red informática. Las

redes y las autopistas de comunicación abrieron nuevas posibilidades para la educación para el desarrollo de un aprendizaje colaborativo, en un modelo de comunicación de muchos a muchos, con posibilidades de una interacción inmediata, donde el profesor asume una concepción diferente del desarrollo del proceso de enseñanza-aprendizaje. Exige un cambio en el pensamiento y la acción del profesor y los estudiantes, que incluye la posibilidad de disponer de recursos tecnológicos como mediadores que posibilitan la interacción y el intercambio de ideas y materiales entre ellos mediante chat, foros, listas de discusión, entre otros.(3)

Hoy en día, fomentar el estudio de idiomas es uno de los principales retos del mundo debido a la variedad de lenguas que existen y a los muchos dialectos que de ellas se derivan. El estudio de idiomas es imprescindible para fomentar las buenas relaciones y la cooperación entre los pueblos de todo el mundo. Para esto en la UCI se asume el concepto de Pat Grounds, especialista del Consejo Británico lo define al Centro de Autoacceso como "Un lugar donde el usuario tiene la oportunidad de seleccionar entre una gran variedad de materiales (impresos, cassettes, videos, multimedia) que le sirven para incrementar y reforzar sus conocimientos en el aprendizaje de un idioma con la ventaja de invertir o dedicar el tiempo que considere adecuado a sus necesidades, además de establecer el ritmo de avance de su aprendizaje. Es también un espacio donde el usuario en caso de requerirlo recibe atención de un asesor académico".

En los Centros de Autoaprendizaje de las Lenguas Extranjeras (CASIE) en la UCI, los estudiantes encuentran un ambiente alternativo de aprendizaje, que les permite trabajar de manera independiente en sus debilidades y profundizar en los temas que son de su interés, mediante un entorno que se adapta a su particular estilo de aprender. Los materiales están orientados a perfeccionar las habilidades básicas en el aprendizaje de idiomas (lectura, comprensión auditiva, redacción, gramática y pronunciación).

La idea de crear un proyecto para la gestión del autoaprendizaje de lenguas extranjeras en la UCI surge por la necesidad de fomentar de una manera más coordinada e inmediata el proceso de autoaprendizaje, teniendo como premisa un cambio en la innovación pedagógica y en la formación de los ingenieros en cuanto al estudio de los idiomas extranjeros. Esto persigue que el estudiante sea capaz de autogestionar su aprendizaje, de aprender de forma autónoma y consciente sin la ayuda de terceros, siguiendo además su propio ritmo de aprendizaje y a partir de sus necesidades, siendo cada vez más el centro del proceso de enseñanza y aprendizaje.

El proyecto CEVALE (Centro Virtual de Autoaprendizaje de Lenguas Extranjeras) surge como respuesta a las necesidades existentes en los CASIE expresadas por el Departamento Central de Idiomas Extranjeros (DDC Idiomas) en la UCI. Está considerado como sistema inteligente capaz de generar rutas de aprendizaje de idiomas. Se espera que este sistema sea montado sobre una plataforma y que genere tantas actividades como tantos recursos existan en este.

Teniendo en cuenta el sistema que se quiere desarrollar la **situación problemática** está dada por la inexistencia en la UCI de un sistema que gestione la inserción, modificación, actualización y eliminación de los recursos, dígase información y ejercicios para el autoaprendizaje de las lenguas extranjeras.

Teniendo en cuenta la explicación anterior de la situación problemática el **problema a resolver** es: ¿Cómo erradicar las dificultades existentes para la gestión de los recursos para el autoaprendizaje de las lenguas extranjeras en la UCI?

El **objeto de estudio** es el proceso de gestión de los recursos en el sistema CEVALE dentro de los CASIE. Planteándose como **objetivo general**:

- Desarrollar el diseño del módulo de gestión de los recursos del sistema CEVALE que permita el eficiente manejo de los recursos dentro de los CASIE.

El **campo de acción** es la informatización de la gestión de los recursos en los CASIE en la UCI y como **idea a defender**: Con el diseño del módulo de gestión de los recursos del sistema CEVALE, se podrán manejar de forma organizada los recursos dentro de los CASIE.

Para cumplir correctamente el objetivo antes mencionado, se han definido las siguientes **tareas de la investigación**:

- ✓ Definir el Diseño teórico de la investigación.
- ✓ Definir el Diseño metodológico de la investigación.
- ✓ Valorar el estado del arte referente a los recursos en plataformas de autoaprendizaje.
- ✓ Caracterizar los procesos y la estructura de los CASIE.
- ✓ Realizar el análisis del módulo de recursos.

- ✓ Seleccionar las herramientas a utilizar para el diseño del módulo.
- ✓ Diseñar el módulo de Recursos.

Se han utilizado diferentes **métodos científicos** para dar cumplimiento de forma exitosa a este proyecto, dentro de los cuales se encuentran:

Métodos teóricos:

- **Analítico – sintético:** Este método tiene como objetivo analizar las teorías, documentos, etc., permitiendo la extracción de los elementos más importantes que se relacionan con el autoaprendizaje y con el proceso de gestión de recursos en aplicaciones Web.
- **Análisis histórico – lógico:** Se analizará y revisará la documentación existente sobre el tema, así como el estado del arte en el desarrollo de aplicaciones encargadas de gestionar recursos en aplicaciones Web, lo que permitirá la profundización del conocimiento en el tema.
- **Modelación:** La modelación es el método mediante el cual se crean abstracciones con el objetivo de explicar la realidad. Se realiza la modelación de distintos diagramas del sistema que se está desarrollando.

Métodos empíricos:

- **Observación:** Se hace uso de este método por la necesidad de conocer aspectos de comportamiento, se realizarán visitas a los CASIE con el objetivo de conocer mejor el sistema de aprendizaje de idiomas en la UCI.
- **Entrevista:** La entrevista es una conversación planificada entre el investigador y el entrevistado para obtener información. Estas se realizarán a los clientes para conocer la realidad de la situación problemática y las oportunidades potenciales para el desarrollo de aplicaciones para la gestión de recursos.

La **población** estará compuesta por los once especialistas de seguridad que elaboran en la UCI, los veintidós especialistas del Departamento Central de Idiomas Extranjeros de la UCI y los nueve integrantes que conforman los otros módulos del sistema. Se toma el 15% como grado de confiabilidad al cual se le realizarán entrevistas para conocer mejor el desarrollo del autoaprendizaje de idiomas en la UCI, por tanto la **muestra** es heterogénea y será de seis especialistas por área y como **técnica de**

muestreo se utiliza la probabilística y se toma la del muestreo estratificado porque subdivide una población heterogénea en una serie de sub-poblaciones homogéneas para garantizar que todas las características de la población heterogénea estén representadas en la muestra.

CAPÍTULO 1: Fundamentación Teórica

1.1 Introducción

En este capítulo se tratan los conceptos fundamentales que ayudan a un mejor entendimiento del problema que se investiga; que en esencia es la no existencia de un sistema que gestione (inserción, modificación, actualización y eliminación) los recursos (información y ejercicios) para el autoaprendizaje de las lenguas extranjeras. Se aborda más ampliamente el objeto de estudio, así como una descripción muy general del entorno donde se desarrolla el problema. Además se estudian otras soluciones existentes teniendo en cuenta las principales características, con el fin de obtener la mejor solución.

1.2 Recursos

Con relación a la cuestión del grado de aplicación de las TICs en la enseñanza hay que indicar que existen actualmente muchos programas de ordenador y páginas Web de carácter educativo, para todas las materias y niveles de enseñanza, pero son todavía muchos los profesores que no los utilizan de modo que la mayoría de tales programas se van quedando obsoletos o anticuados sin llegar a aplicarse en contextos educativos reales y sin evaluar su posible utilidad didáctica.

En los últimos tiempos esta situación está cambiando. Por una parte se están desarrollando sistemas y lenguajes de autor, que facilitan el diseño de unidades didácticas por parte de los profesores interesados en esta temática. Mediante un sistema de autor se pueden elaborar lecciones, incluyendo diagramas, gráficos, imágenes, textos, cuestiones y permitiendo la evaluación de las respuestas, mediante la realimentación adecuada. Estas herramientas permiten al profesor mayor libertad en la estructura de los temas pero también requieren bastante dedicación.

Por otra parte, los lenguajes y sistemas de autor están evolucionando rápidamente en conexión con los avances sobre tutores inteligentes y sistemas adaptativos multimedia, que están mejorando notablemente las posibilidades de interacción del alumno con los programas, el acceso a la información, la presentación de contenidos y el diseño de actividades de aprendizaje. Esta panorámica abre unas importantes expectativas para la evolución futura de la enseñanza asistida por ordenador,

ya que permite diseñar sistemas tutoriales flexibles y adaptados al nivel de conocimientos de cada alumno.

Tipos de recursos informáticos que puede utilizar el profesorado

Con relación a la cuestión de los tipos de recursos informáticos que puede utilizar el profesorado y las posibles aplicaciones educativas de los diferentes recursos, hay que hacer una distinción entre los recursos informáticos de propósito general y los programas específicos de enseñanza asistida por ordenador.

Recursos informáticos:

- Procesadores de texto
- Bases de datos
- Hojas de cálculo
- Diseño de presentaciones
- Entornos de diseño gráfico
- Navegadores de Internet
- Gestores de correo electrónico
- Diseño de páginas Web
- Programas de ejercitación y autoevaluación
- Tutoriales interactivos
- Enciclopedias multimedia
- Simulaciones y laboratorios virtuales
- Laboratorio asistido por ordenador
- Tutores inteligentes
- Sistemas adaptativos multimedia
- Sistemas de autor

Recursos informáticos de propósito general

Se denominan programas de propósito general a las aplicaciones informáticas que pueden ser útiles para todo tipo de usuarios de ordenador, entre las que actualmente se destacan las llamadas herramientas de ofimática tales como procesadores de texto (Word, Word Perfect), bases de datos

(DBase, Acces), hojas de cálculo (Excel), presentaciones (Harvard Graphics, Power Point), entornos de diseño gráfico (Paint, Corel Draw, Autocad) y otros tipos de herramientas como los navegadores de Internet (Explorer, Nestcape), gestores de correo electrónico (Outlook Express) y recursos para la edición y diseño de páginas Web (FrontPage).

Con relación a estas aplicaciones de propósito general hay que señalar que, aunque no tienen necesariamente un carácter educativo, es sumamente conveniente que el profesorado de todos los niveles tenga un conocimiento adecuado de algunas de estas aplicaciones, con el fin de poder utilizarlas en diversas actividades relacionadas con la enseñanza. No es necesario a estas alturas reflejar la importancia educativa de manejar un procesador de textos para que el profesor pueda elaborar apuntes, actividades de clase, exámenes, etc. También es bastante recomendable para el profesorado aprender a elaborar y utilizar en clase presentaciones didácticas por ordenador, o bien colecciones de diapositivas y transparencias, usando una herramienta bastante generalizada y sencilla de usar como Power Point. Por último, no se puede olvidar la gran importancia desde el punto de vista cultural y educativo de saber manejar un navegador para buscar información de todo tipo en Internet o el manejo del correo electrónico para comunicarse entre profesores y alumnos. Por tanto, los citados recursos deberían formar parte de la formación mínima que todo profesor debería adquirir en relación al uso de las TICs en la enseñanza. Los otros tipos de programas de propósito general (bases de datos, hojas de cálculo, diseño de gráficos y de páginas web), aunque pueden tener un uso más restringido y específico que los anteriores, también pueden ser útiles en la docencia y, por tanto, debería favorecerse su conocimiento entre el profesorado.(4)

Características de los recursos:

- Admite la presentación de cualquier contenido digital, Word, PowerPoint, Flash, vídeo, sonidos, etc.
- Los archivos pueden subirse y manejarse en el servidor, o pueden ser creados sobre la marcha usando formularios Web (de texto o HTML).
- Se pueden enlazar contenidos externos en Web o incluirlos perfectamente en la interfaz del curso.
- Pueden enlazarse aplicaciones Web, transfiriéndoles datos.

Recursos didácticos

Los recursos didácticos agrupan todos los objetos, aparatos, medios de comunicación que pueden ayudar a descubrir, entender o consolidar conceptos fundamentales en las diversas fases del aprendizaje. Se incluye en su clasificación todo tipo de material, ya sean estos software didácticos y no didácticos, libros, juegos, notaciones simbólicas, representaciones gráficas y, en general, todas las formas expresivas e instrumentales que permitan el trabajo docente.(5)

Las investigaciones han demostrado que el nivel y el alcance de los recursos didácticos que pueden usar los docentes de la Universidad se sustentan en la experiencia, sin mucha construcción de conceptos. Por lo tanto, los recursos deben ser sencillos para facilitar que haya un primer acercamiento a la noción. El objetivo final de todo este proceso es fomentar y sostener un aprendizaje activo que haga de los estudiantes verdaderos actores del proceso de adquisición de conocimientos, en el que ellos se involucran en el hacer y en el pensar sobre lo que aprenden.(5)

Los medios y recursos didácticos pueden considerarse como herramientas de ayuda para llevar a cabo la tarea formativa, siempre que se haga un uso correcto y adecuado de ellos.

Al igual que se encuentran una gran variedad de definiciones acerca de lo que se entiende por medios y recursos didácticos, existen también muchas maneras de clasificarlos. Con el objetivo de facilitar su estudio y elección se han optado por hacer una división en tres grandes grupos:

- Medios tradicionales.
- Medios audiovisuales.
- Nuevas tecnologías.

En función de quién sea el encargado de elaborar estos medios se diferencian entre:

- Diseñados por el formador y/o los alumnos.
- Diseñados por profesionales de la producción.

Los medios elaborados por el formador se caracterizan por tener un alto valor educativo, que viene dado fundamentalmente por el proceso seguido en su diseño y producción. De esta forma, se anima a

que los formadores se atrevan a construir y utilizar medios artesanos propios adaptados a sus situaciones concretas de formación. Sin embargo, es necesario tener en cuenta algunos criterios en el diseño y elaboración de los medios didácticos.(6)

Los medios didácticos deben apoyar y facilitar la labor docente, no entorpecerla. Por esta razón, no es preciso la elaboración de medios muy sofisticados que el formador tenga que dedicar mucho tiempo y esfuerzo al manejarlos.

- Adecuación a los objetivos perseguidos. Los medios deberán construirse teniendo siempre presente las funciones que van a desempeñar.
- Adecuación a las necesidades y características de los alumnos, (nivel, contexto, etc.). Elaborar los propios medios facilita tener en cuenta estos requisitos.(6)

1.3 Gestionar recursos sobre un sistema web

El proyecto del Centro de Recursos Virtuales (CRV) provee una estructura que se puede usar indistintamente desde Internet o en entornos de redes intranet, como los que ha instalado el MEP a través de PROMECE o de Fundación Omar Dengo en diversas instituciones costarricenses. Esta estructura permite al docente administrar materiales para sus estudiantes, el diseño de planes de estudio y el seguimiento individualizado de algunas de las tareas que el estudiante desarrolle dentro del sistema. También permite al estudiante tener acceso a una serie de materiales desarrollados para apoyar su estudio.

Las funcionalidades claves del sistema administrativo son:

- Gestión de materiales didácticos: publicar, eliminar, modificar.
- Gestión de Preguntas: Pruebas con retroalimentación en tiempo real.
- Administración de usuarios.
- Administración de Centros.
- Administración de Programas.
- Otras funciones periféricas.

Un aspecto esencial de este proyecto lo constituyen los materiales: están a disposición de los estudiantes y docentes una serie de documentos elaborados bajo estándares de formato, calidad y forma que permiten proveer recursos didácticos a los estudiantes.

La filosofía general del CRV está sustentada en brindar una base de recursos didácticos ya disponibles, las cuales se complementarán con materiales publicados por los propios docentes o profesionales que quieran colaborar. En general el procedimiento para agregar materiales a las bases de datos o nuevas preguntas es bastante simple.

La mayoría de las veces, el tipo de recursos que se presenta está bastante lejano a los materiales que un profesor puede encontrar en un libro; la verdad es que la diversidad y versatilidad que este sistema ofrece supera por mucho lo que se puede obtener de un libro de texto. Los materiales y los generadores de pruebas permiten al docente y al estudiante planificar diversos niveles de profundidad en la exploración de la materia, desde niveles muy instrumentales hasta tareas que implican razonamiento abstracto.

1.4 Sistema Web de Apoyo a la Docencia (SWAD)

La utilización de portales Web de apoyo al aprendizaje, a la docencia y a la gestión de datos de los alumnos de una universidad es interesante porque facilita al profesor la realización de algunas tareas docentes y de gestión, por ejemplo, la difusión de documentos o la consulta de datos de los alumnos, pero sobre todo porque propicia la autonomía y el autoaprendizaje del estudiante, ya que éste puede hacer uso de tales servicios en cualquier instante y desde cualquier sitio.

Debido a la complejidad de implementación de los servicios web dinámicos, no es lógico que distintos profesores realicen la programación de los mismos para sus respectivas asignaturas. Es mucho más interesante que cualquier profesor de una titulación, centro, o universidad, disponga de una herramienta común personalizable según sus necesidades o preferencias.

En los últimos años ha comenzado a extenderse el uso de plataformas de E-learning y B-learning tanto comerciales como de libre distribución que aglutinan los servicios software necesario para dar soporte a una infraestructura completa de teleformación a través de Internet. En general, estos sistemas permiten la creación de cursos o asignaturas que pueden diseñarse y gestionarse como un todo por personal docente, sin necesidad de un conocimiento muy técnico. Sus principales inconvenientes son el elevado coste de adquisición (en el caso de las plataformas comerciales), la

dependencia de una compañía externa, y la dificultad de adaptación a las necesidades y particularidades de cada organización docente.

Objetivos

Los objetivos planteados al abordar el desarrollo de la plataforma SWAD pueden concretarse según los posibles beneficiarios de su aplicación:

- Para profesores y otros administradores de la plataforma, los objetivos propuestos han sido la realización a través de Internet de la mayoría de las tareas de gestión relacionadas con una asignatura y sus alumnos, y la mejora en la tutorización y comunicación en general con los mismos.
- Para los alumnos, los objetivos planteados han sido la mejora del acceso a materiales e información de las asignaturas, la posibilidad de autoevaluación a distancia, y la ampliación de los medios de comunicación tanto entre unos alumnos y otros, como entre alumnos y profesores.

El principal criterio contemplado en el desarrollo de la plataforma ha sido facilitar su utilización en cualquier asignatura impartida, haciendo hincapié tanto en la facilidad de aprendizaje y uso para estudiantes y profesores, como en el ahorro de tiempo y mejora de calidad en diversas tareas propias de la docencia.

Servicios ofrecidos

La plataforma SWAD pretende alcanzar los objetivos propuestos ofreciendo los siguientes servicios:

- Información y documentación de asignaturas
- Calendario académico
- Horario
- Bibliografía, FAQ, enlaces
- Descarga de archivos (documentos, transparencias, relaciones de problemas, etc., aunque sin creación de contenidos HTML)
- Evaluación de estudiantes
- Información sobre el sistema de evaluación
- Ficha personalizada del alumno en la asignatura

- Exámenes de autoevaluación de tipo test
- Información y administración de estudiantes y profesores
- Envío de correo electrónico
- Foros
- Pizarra / Chat
- Estadísticas y control de accesos
- Consulta por usuario, fecha, hora, acción, titulación, asignatura. (4)

1.5 Informatización de los procesos de gestión de los recursos dentro de los CASIE de la UCI

En los Centros de Autoaprendizaje y Servicios de Idiomas Extranjeros de la de la Universidad de las Ciencias Informáticas los usuarios asisten a ella en busca de recursos que le ayuden en su autoaprendizaje ya sean de información o de ejercicios. Por esto se hace necesario que los recursos solicitados tengan buena calidad, acceder con facilidad al recurso solicitado y que haya una gran variedad de ellos, dándole así al usuario varias opciones para enriquecer su conocimiento. Dada esta situación el presente trabajo se basa en la informatización de este proceso de gestión de recursos en los CASIE.

1.5.1 Centros virtuales de recursos en la UCI

Los Centros Virtuales de Recursos son espacios en la red diseñados especialmente para ofrecer a alumnos y profesores las mayores facilidades para conocer y utilizar los recursos disponibles y la producción de otros, facilitando la interacción y el trabajo colaborativo. En el mismo se brindan además servicios de información y facilidades de trabajo colaborativo, gestión de recursos e intercambio entre estos.

La naturaleza de los recursos didácticos a gestionar en este tipo de centro se refiere a todos aquellos materiales con un diseño didáctico previo que apoyan el proceso de enseñanza aprendizaje y que contribuyen al desarrollo de la competencia comunicativa en el idioma inglés, en correspondencia con una concepción pedagógica definida.

El trabajo que se realiza con los recursos en el mismo incluye la evaluación y validación de los recursos que existen actualmente en la red a través de la investigación y el debate sobre cada uno de ellos; el registro, clasificación, catalogación y ordenamiento de los recursos; la selección de recursos

de acuerdo a las necesidades y características de la institución y los usuarios y la evaluación de la efectividad de los recursos utilizados y la producción de materiales didácticos.

Poder contar con un CVR con gran cantidad de recursos clasificados y con valores añadidos a partir de un sistema de evaluación de los mismos, la posibilidad de utilizar herramientas de comunicación síncrona y asíncrona y para el trabajo colaborativo permitió el desarrollo de un grupo de competencias fundamentales en estudiantes y profesores de idiomas, así como un gran caudal de recursos que se comparten y actualizan constantemente de forma colaborativa.

1.5.2 Sistemas para la gestión de autoaprendizaje vinculados al campo de acción

Los Centros de Autoaprendizaje y Servicios de Idiomas Extranjeros en la Universidad de las Ciencias Informáticas brindan a sus usuarios (estudiantes, profesores y la comunidad universitaria en general) una nueva variante para el autoaprendizaje de idiomas extranjeros. Los CASIE son espacios físicos y virtuales los cuales benefician la autogestión del aprendizaje por partes de los clientes de los mismos. Su propósito es proporcionar al cliente todos los recursos necesarios para estudiar la lengua que desee como por ejemplo: recursos didácticos, tecnologías y asesorías a través de entornos de aprendizaje altamente efectivo. La filosofía del CASIE se basa en los siguientes principios:

- se centra en el estudiante (ya que cada persona tiene una manera de aprender diferente a los demás, donde los objetivos, la velocidad de aprendizaje, las motivaciones e intereses son particularmente de naturaleza personal).
- en la personalización, individualización y en la flexibilidad del proceso de enseñanza-aprendizaje.
- en la comunicación y la interrelación de toda la comunidad educativa.
- en el trabajo cooperativo en ambientes interactivos de aprendizaje.
- en la autogestión de los recursos de aprendizaje.
- en la solidaridad ciberespacial y el respeto mutuo.
- en la innovación.
- en el aprender a aprender de forma constante.

Los objetivos principales de los CASIE en la UCI son:

- Proporcionar a la comunidad universitaria un espacio de estudio que complemente el esquema de ofertas para el aprendizaje de idiomas, permitiendo de acuerdo con las necesidades y motivación del cliente, avanzar y reforzar su aprendizaje con un ritmo propio, contando con una asesoría por parte del personal especializado de los CASIE y contando a su vez con una amplia gama de recursos didácticos y tecnológicos para el autoaprendizaje de idioma extranjeros.
- Ayudar a los estudiantes a desarrollar sus competencias de aprendizaje mediante las sesiones de tutoría con el propósito de que tomen responsabilidad de su propio aprendizaje y les facilite el camino a la autonomía.
- Involucrar a las personas que forman parte de la comunidad universitaria que por problemas de horario o por lo especial de sus necesidades para aprender idioma no asisten a clases presenciales.
- Ofrecer además otros servicios (traducción e interpretación, copiado de CD y casetes de audio y video, escaneado de documentos, etc.) que apoyen el aprendizaje de idiomas extranjeros en la universidad.

1.6 Descripción actual del dominio del problema

En estos momentos el Centro de Autoaprendizaje y Servicios de Idiomas Extranjeros recibe cada día usuarios que llegan a este centro para el aprendizaje de idiomas extranjeros. De los mismos se recogen sus datos generales y luego se presentan a realizar una encuesta y de acuerdo al resultado obtenido este llena un contrato donde se plasman sus datos personales, los estilos de aprendizaje, los horarios y la forma de evaluación con el cual va a realizar su aprendizaje, luego todos estos datos son archivados en un registro de usuario y todas las actividades que son realizadas por él mismo. Además de llevar en el Registro de Consulta de Libro y en el Registro de Usuario los recursos que son utilizados por el cliente ya sean de tipo digital o físico.

1.7 Situación problemática

En la Universidad de las Ciencias Informáticas, actualmente EVA (Entorno Virtual de Aprendizaje) presenta algunas dificultades que no satisfacen todas las necesidades del modelo pedagógico utilizado

en la disciplina de Idiomas Extranjeros y otras dadas por el nivel real en el dominio del idioma que presenta el usuario. Entre ellas a modo de ejemplo se pueden citar (CEVALE):

- No está disponible un sistema informatizado en línea las 24 horas, en el cual, sin la intervención de un profesor (tutor, guía o asesor físico) el usuario pueda completar una ruta de aprendizaje; elemento distintivo éste, que no es posible con el EVA actual.
- No incorpora nuevas modalidades o tipologías de recursos a utilizar en la enseñanza de las lenguas extranjeras, no posible en el actual EVA.
- No desarrolla un sistema automatizado que brinde la posibilidad de, dado un conjunto de diagnósticos y test iniciales respondidos por el usuario, automáticamente procese dicha información y determine el estado actual del usuario así como su estilo o estilos de aprendizaje y genere la ruta de autoaprendizaje a seguir por éste durante su entrenamiento.
- No desarrolla un sistema sustentado en la comunicación con el actual EVA, así como con otras futuras plataformas a utilizar teniendo en cuenta el modelo pedagógico seleccionado, respetando los estándares internacionales en el desarrollo de este tipo de aplicación.
- No posibilita la atención a una población potencial de más de 12 mil usuarios a través de los CASIE.

La problemática fundamental radica en lograr la formación de habilidades y conocimientos teniendo en cuenta los diferentes niveles de aprendizaje de las lenguas extranjeras, en los que se puede encontrar el usuario, sustentado en un modelo pedagógico no presencial (a distancia), y elaborar un sistema informático para tal fin.

1.8 Propuesta de solución

El sistema Web contará con diferentes tipos de recursos: información o ejercicios los que pueden estar dados en libros, multimedia, casetes de audio, casetes de video, revistas, diccionarios, tareas docentes.

Con la gestión de estos recursos por parte de los especialistas le permitirá al usuario mostrar un mayor interés en su autoaprendizaje ya que tendrá recursos según su interés de conocimiento y podrán además elegir del tema que deseen aprender uno o varios recursos.

1.9 Análisis de otras soluciones existentes

➤ Joomla

Sistema gestor de contenidos que permite crear sitios web de alta interactividad, profesionalidad y eficiencia. La administración de Joomla está enteramente basada en la gestión online de contenidos, gestión online porque todas las acciones que realizan los administradores de sitios Joomla, ya sea para modificar, agregar, o eliminar contenidos se realiza exclusivamente mediante un navegador web (browser) conectado a Internet, es decir, a través del protocolo HTTP (Protocolo de transferencia de hipertexto).

Con Joomla se permite crear sitios web de noticias, sitios corporativos, sitios web de presencia, portales comunitarios, e incluso también puede crearse con Joomla sistemas que funcionen en redes cerradas (Intranets) para gestionar información interna (comunicaciones, usuarios, etc.) de compañías o empresas de negocios.

Es tan sencilla y amigable la interfaz administrativa de Joomla, que cualquier persona puede administrar sus propios contenidos web sin la necesidad de poseer conocimientos técnicos, sin saber lenguaje HTML, y sin recurrir a un WebMaster cada vez que hay que actualizar alguna cosa en un sitio web. Todos los aspectos técnicos del sistema están ocultos al usuario común, y han sido resueltos por los programadores de Joomla, haciendo las cosas sencillas para que los administradores y editores de contenidos puedan prescindir casi completamente de una capacitación técnica especializada.

Joomla está programado en lenguaje PHP (Hypertext Pre Processor) y SQL (Structure Query Language). Utiliza bases de datos relacionales, más específicamente MySQL. Tanto PHP como MySQL son programas OpenSource de libre distribución y uso, y al ser Joomla una aplicación web, funciona obviamente en servidores de páginas web (HTTP Servers). Estos servidores de páginas web pueden ser de pruebas o también servidores comerciales de producción.

Basados en esta misma filosofía OpenSource se puede afirmar también que Joomla correrá mejor en datacenters cuyas plataformas sean Linux/Unix y cuyos servidores HTTP sean Apache. Con esto no se quiere decir que Joomla no funciona en plataformas Windows con servidores IIS (Internet Information Server, de Microsoft), pero sí que lo hace con toda su potencia y funcionalidades en plataformas Unix/Apache.

Si bien Joomla corre mejor en servidores Unix/Apache, la administración y edición de contenidos desde la computadora que accede el administrador puede funcionar tanto con sistemas Unix o Windows.(7)

➤ e107

Sistema de Manipulación de Contenidos escrito en PHP y usa MySQL como base de datos. Es completamente gratuito y está en constante desarrollo, y goza de una gran cantidad de opciones y herramientas prácticas que permiten personalizar a gusto los proyectos gestionados con esta aplicación. Con un perfil bajo pero con grandes características se convierte en una opción muy interesante para quienes quieren utilizar un CMS por primera vez ya que tanto la instalación como la administración del sitio es muy sencilla e intuitiva.

Lamentablemente e107 no ha logrado popularidad en las comunidades de habla hispana y es por eso que los recursos y el soporte técnico encontrado generalmente son en inglés y en otros idiomas.

Características:

- Inspector de archivos que permite verificar archivos antiguos, innecesarios, corruptos, dañados o que no correspondan al sistema.
- Completo panel de administración asistido con referencias breves.
- Importante cantidad de plugins incluidos en el pack oficial.
- Multiposición de bloques.
- Gestión de permisos por grupo.(8)

➤ Moodle

Sistema de gestión de cursos de libre distribución que ayuda a los educadores a crear comunidades de aprendizaje en línea. Fue diseñado por Martin Dougiamas de Perth, Australia Occidental, apoyándose en el marco de la teoría del constructivismo social. El Sr. Dougiamas tiene un interesante background tecnológico pero también lo combina con su reciente carrera en educación. Como parte de éste, desarrolló Moodle, basado en su conocimiento sobre la teoría del aprendizaje y la colaboración.

Además de la característica del código libre ya mencionada, algunas de las ventajas que más valoran los docentes son: su filosofía constructivista, su estructura modular, su amplia comunidad de desarrolladores y la gran cantidad de documentación.

La teoría del constructivismo social aplicada a Moodle se basa en que el conocimiento se adquiere cuando los miembros de una comunidad interaccionan entre sí y son a la vez creadores y receptores de información. La colaboración entre profesor-alumno transforma al alumno en protagonista del proceso del aprendizaje. El diseño y el desarrollo de Moodle está basada esta filosofía.

La solidez de la aplicación es otra de sus ventajas debido a la combinación de un lenguaje de programación PHP y a la base de datos relacional MySQL. Ambos elementos permiten una estructura modular que facilita su uso y aprendizaje ya que se van utilizando partes según se van conociendo. Su utilización es bastante sencilla y las operaciones básicas se realizan sin dificultad de manera intuitiva.

Su gran popularidad le lleva a contar con una red muy amplia de desarrolladores que conforman una comunidad de más de 130.000 usuarios registrados en 160 países y 75 idiomas. Los usuarios más avanzados se han dedicado a publicar, en la página oficial de Moodle, manuales y guías online. Son documentos muy completos que facilitan el uso y la adaptación de los usuarios a esta plataforma.

Algunas características:

- Gran disponibilidad: satisface las necesidades de profesores, estudiantes, administradores y creadores de contenidos.
- Escalabilidad: la aplicación se adapta a las necesidades que aparecen en el transcurso de la utilización de la misma. Tanto en organizaciones pequeñas como grandes se pueden utilizar la arquitectura Moodle.
- Facilidad de uso: Las utilidades de Moodle son sencillas y su utilización es muy intuitiva. Existen manuales de ayuda que facilitan su utilización.
- Interoperabilidad: el código abierto propicia el intercambio de información gracias a la utilización de los "estándares abiertos de la industria para implementaciones web" (SOAP, XML) Además se puede ejecutar en Linux, MacOS y Windows.
- Estabilidad: Moodle es un entorno eficaz y confiable.
- Seguridad: la restricción de acceso a las comunidades de aprendizaje de Moodle es una solución para evitar riesgos innecesarios.(9)

Módulos principales en Moodle

- Módulo de Tareas
- Módulo de Consulta
- Módulo Foro
- Módulo Diario
- Módulo Cuestionario
- Módulo Material
- Módulo Encuesta

Moodle permite varios tipos de recursos:

- Editar una página de texto: es una simple página escrita en texto plano.
- Editar una página Web: facilita la elaboración de una página web completa dentro de Moodle a través de un editor HTML, la cual se almacena en la semana o tema donde se esté trabajando.
- Enlazar un archivo o una Web: permite enlazar páginas, archivo de la Web o archivos personales. Las páginas Web normales se muestran tal como son, en tanto que los archivos multimedia se tratan de modo inteligente y pueden incrustarse dentro de una página web. Por ejemplo, los archivos MP3 pueden mostrarse utilizando un reproductor incorporado; igualmente sucede con los archivos de video, animaciones flash, etc. Existen varias opciones para mostrar contenidos: ventanas emergentes, ventanas con marcos, entre otras.
- Mostrar contenido IMS: permite agregar a su curso paquetes que se ajustan a las especificaciones de contenido de paquete IMS. Simplemente, se debe subir un paquete comprimido y desplegarlo. Además, mostrar un directorio completo, incluyendo los subdirectorios, desde el área de archivos de su curso. Los estudiantes pueden ver todos los archivos y navegar a través de ellos.
- Mostrar un directorio: puede mostrar un directorio completo junto con sus subdirectorios, desde el área de archivos de su curso. Los estudiantes pueden ver todos los archivos y navegar por ellos.
- Añadir una etiqueta: las etiquetas son un tipo diferente de recurso por cuanto permite poner títulos, intertítulos, textos cortos o imágenes que se visualizan directamente en cada semana o tema en la página principal del curso.(4)

Ventajas:

Una de las características más atractivas de Moodle, que también aparece en otros gestores de contenido educativo, es la posibilidad de que los alumnos participen en la creación de glosarios, y en todas las lecciones se generan automáticamente enlaces a las palabras incluidas en estos.

Desventajas:

Algunas actividades pueden ser un poco mecánicas, dependiendo mucho del diseño instruccional para Moodle lo que lo hace más dinámico y atractivo. Por estar basado en tecnología PHP la configuración de un servidor con muchos usuarios debe ser cuidadosa para obtener el mejor desempeño. (10)

Teniendo en cuenta las características de estos Sistemas de Manipulación de Contenidos y las grandes ventajas que ofrece Moodle, se concluye que es el más seguro y confiable para el desarrollo de la investigación que se realiza.

1.10 Conclusiones parciales

Actualmente, el proceso de control de las actividades que se desarrollan en los CASIE con el fin de fomentar el autoaprendizaje de sus usuarios es ineficiente, debido a las necesidades existentes en éstos. Un problema fundamental se debe a la gran cantidad de usuarios que solicitan sus servicios, pues todo este proceso se realiza de forma manual, provocando que sea lento y que no se cuente con el control necesario de los usuarios. En este capítulo se realizó un estudio de los principales conceptos asociados al problema existente así como el estudio de otras soluciones, con el objetivo de reunir sus mejores características, que contribuyen a un mejor desarrollo de la investigación.

CAPÍTULO 2: Tendencias y tecnologías actuales a desarrollar

2.1 Introducción

El objetivo de este capítulo es la fundamentación de las tecnologías que se van a utilizar en esta investigación, temas relacionados con las herramientas, lenguaje y gestor de base de datos a utilizar, así como el lenguaje de modelado y la metodología a emplear, además se describen temas asociados con la seguridad en aplicaciones web, igualmente temas conectados con el sistema operativo a utilizar, teniendo en cuenta las características y las necesidades planteadas por el Departamento Central de Idiomas Extranjeros y los Centros de Autoaprendizaje y Servicios de Idiomas Extranjeros en la Universidad de las Ciencias Informáticas.

2.2 Metodología de desarrollo de software

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software.

Es como un libro de recetas de cocina, en el que se van indicando paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando además, qué personas deben participar en el desarrollo de las actividades y qué papel deben tener. Además detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.

Actualmente es imprescindible considerar los riesgos, aunque habitualmente las empresas, no han sido concienciadas de los riesgos inherentes al procesamiento de la información mediante ordenadores, a lo que han contribuido, a veces, los propios responsables de informática, que no han sabido explicar con la suficiente claridad las consecuencias de una política de seguridad insuficiente o incluso inexistente. Por otro lado, debido a una cierta deformación profesional en la aplicación de los criterios de coste/beneficio, el directivo desconocedor de la informática no acostumbra a autorizar inversiones que no lleven implícito un beneficio demostrable, tangible y mensurable.

Las técnicas indican cómo debe ser realizada una actividad técnica determinada identificada en la metodología. Combina el empleo de unos modelos o representaciones gráficas junto con el empleo de unos procedimientos detallados. Se debe tener en consideración que una técnica determinada puede

ser utilizada en una o más actividades de la metodología de desarrollo de software. Además se debe tener mucho cuidado cuando se quiere cambiar una técnica por otra.(11)

Algunas Metodologías:

➤ Feature Driven Development (FDD)

FDD es una metodología ágil diseñada por Peter Coad, Eric Lefebvre y Jeff DeLuca. Se basa en un proceso iterativo con iteraciones cortas que producen un software funcional que el cliente y la dirección de la empresa pueden ver y monitorear. Define claramente entregas tangibles y formas de evaluación del progreso del proyecto. Ayuda a contrarrestar situaciones como el exceso en el presupuesto, fallas en el programa o el hecho de entregar menos de lo deseado. Propone tener etapas de cierre cada dos semanas. Se obtienen resultados periódicos y tangibles.

El mismo pasa por un proceso de cinco pasos secuenciales durante los cuales se diseña y se construye el sistema:

- Desarrollo de un modelo global.
- Construcción de una lista de funcionalidades.
- Planeación por funcionalidad.
- Diseño por funcionalidad.
- Construcción por funcionalidad.(12)

➤ Extreme Programming (XP)

Metodología ágil basada en cuatro principios: simplicidad, comunicación, retroalimentación y valor. Además, orientada por pruebas y refactorización, se diseñan e implementan las pruebas antes de programar la funcionalidad, el programador crea sus propios tests de unidad.



Figura 1: Metodología XP

Este método es típicamente atribuido a Kent Beck, Ron Jeffries y Ward Cunningham. El objetivo de XP son grupos pequeños y medianos de construcción de software en donde los requisitos aún son muy ambiguos, cambian rápidamente o son de alto riesgo. XP busca la satisfacción del cliente tratando de mantener durante todo el tiempo su confianza en el producto. Además, sugiere que el lugar de trabajo sea una sala amplia, si es posible sin divisiones (en el centro los programadores, en la periferia los equipos individuales). Una ventaja del espacio abierto es el incremento en la comunicación y el proporcionar una agenda dinámica en el entorno de cada proyecto.(13)

2.2.1 Metodología a utilizar (RUP)

RUP (Proceso Unificado de Rational) es un proceso para el desarrollo de un proyecto de un software que define claramente quién, cómo, cuándo y qué debe hacerse en el proyecto.

Como 3 características esenciales:

1. Dirigido por los Casos de Uso: que orientan el proyecto a la importancia para el usuario y lo que este quiere.
2. Centrado en la arquitectura: que relaciona la toma de decisiones que indican cómo tiene que ser construido el sistema y en qué orden.
3. Iterativo e incremental: donde divide el proyecto en mini proyectos donde los casos de uso y la arquitectura cumplen sus objetivos de manera más depurada.

Como filosofía RUP maneja 6 principios claves:

1. **Adaptación del proceso:** El proceso deberá adaptarse a las características propias de la organización. El tamaño del mismo, así como las regulaciones que lo condicionen, influirán en su diseño específico. También se deberá tener en cuenta el alcance del proyecto.
2. **Balancear prioridades:** Los requerimientos de los diversos inversores pueden ser diferentes, contradictorios o disputarse recursos limitados. Debe encontrarse un balance que satisfaga los deseos de todos.
3. **Colaboración entre equipos:** El desarrollo de software no lo hace una única persona sino múltiples equipos. Debe haber una comunicación fluida para coordinar requerimientos, desarrollo, evaluaciones, planes, resultados, etc.
4. **Demostrar valor iterativamente:** Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto así como también los riesgos involucrados.
5. **Elevar el nivel de abstracción:** Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software, lenguajes 4GL o esquemas (frameworks) por nombrar algunos. Éstos se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo con UML.
6. **Enfocarse en la calidad:** El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción.

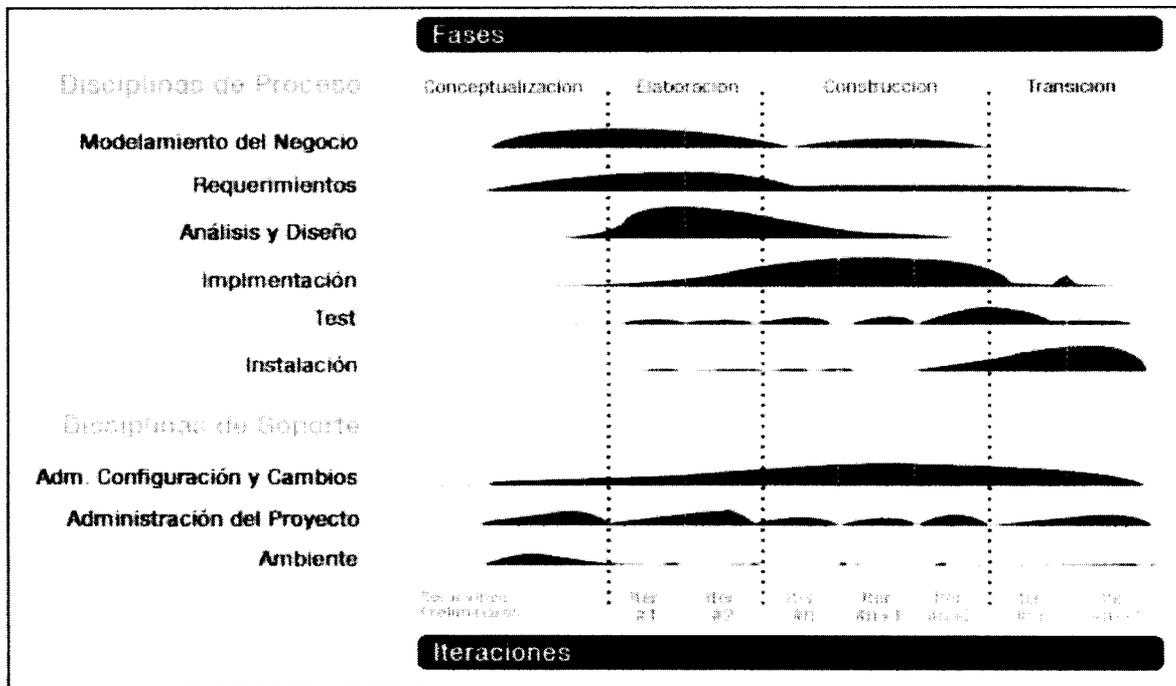


Figura 2: Vista general de RUP

RUP divide el proceso en 4 fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en los distintas actividades.

Fases de RUP:

- Inicio: Se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos. Se define el alcance del proyecto
- Elaboración: se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos
- Construcción: se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario
- Transición: se instala el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados.(14)

Actividades definidas por RUP en cada fase del proyecto:

- Modelado del negocio
- Análisis de requisitos
- Análisis y diseño
- Implementación
- Test
- Distribución
- Gestión de configuración y cambios
- Gestión del proyecto
- Gestión del entorno

RUP es un proceso muy general y grande, por lo que antes de usarlo habrá que adaptarlo a las características de la empresa.(15)

2.3 Lenguaje para el Modelado de Objetos

El lenguaje de modelado de objetos es un conjunto estandarizado de símbolos y de modos de disponerlos para modelar (parte de) un diseño de software orientado a objetos.

Algunos metodólogos del software orientado a objetos distinguen tres grandes "generaciones" cronológicas de técnicas de modelado de objetos:

- En la primera generación, tecnólogos aislados y grupos pequeños desarrollaban técnicas que resolvían problemas que se encontraban de primera mano en los proyectos de desarrollo orientado a objetos. En esta generación se incluye a autores y técnicas como Rumbaugh, Jacobson, Booch, los métodos formales, Shlaer-Mellor y Yourdon-Coad.
- En la segunda generación se reconoció que muchas de las mejores prácticas pertenecían a diferentes métodos del fragmentado terreno de la metodología orientada a objetos. Se realizaron múltiples intentos para integrar dichas técnicas en marcos coherentes tales como FUSION. En cualquier caso, la comunidad del software orientado a objetos empezaba a reconocer los beneficios que la estandarización de las técnicas conllevaría: abandono de las buenas formas de hacer las cosas en favor de la manera adecuada, que permitiría un lenguaje y unas prácticas comunes entre los diferentes desarrolladores.

- La tercera generación consiste en intentos creíbles de crear dicho lenguaje unificado por la industria, cuyo mejor ejemplo es UML (Lenguaje Unificado de Modelado).(16)

2.3.1 Lenguaje Unificado de Desarrollo

UML es un lenguaje para especificar, construir, visualizar y documentar los artefactos de un sistema de software orientado a objetos (OO). Un artefacto es una información que es utilizada o producida mediante un proceso de desarrollo de software.

UML se quiere convertir en un lenguaje estándar con el que sea posible modelar todos los componentes del proceso de desarrollo de aplicaciones. Sin embargo, hay que tener en cuenta un aspecto importante del modelo: no pretende definir un modelo estándar de desarrollo, sino únicamente un lenguaje de modelado. Otros métodos de modelaje como OMT (Object Modeling Technique) o Booch sí definen procesos concretos.

El UML es una técnica de modelado de objetos y como tal supone una abstracción de un sistema para llegar a construirlo en términos concretos. El modelado no es más que la construcción de un modelo a partir de una especificación. Un modelo es una abstracción de algo, que se elabora para comprender ese algo antes de construirlo. El modelo omite detalles que no resultan esenciales para la comprensión del original y por lo tanto facilita dicha comprensión.

Artefactos para el Desarrollo de Proyectos

Un artefacto es una información que es utilizada o producida mediante un proceso de desarrollo de software. Pueden ser artefactos un modelo, una descripción o un software. Los artefactos de UML se especifican en forma de diagramas, éstos, junto con la documentación sobre el sistema constituyen los artefactos principales que el modelador puede observar.

Se necesita más de un punto de vista para llegar a representar un sistema. UML utiliza los diagramas gráficos para obtener estos distintos puntos de vista de un sistema:

- Diagramas de Casos de Uso.
- Diagramas de Clases.
- Diagramas de Comportamiento o Interacción.

- Diagramas de Implementación.

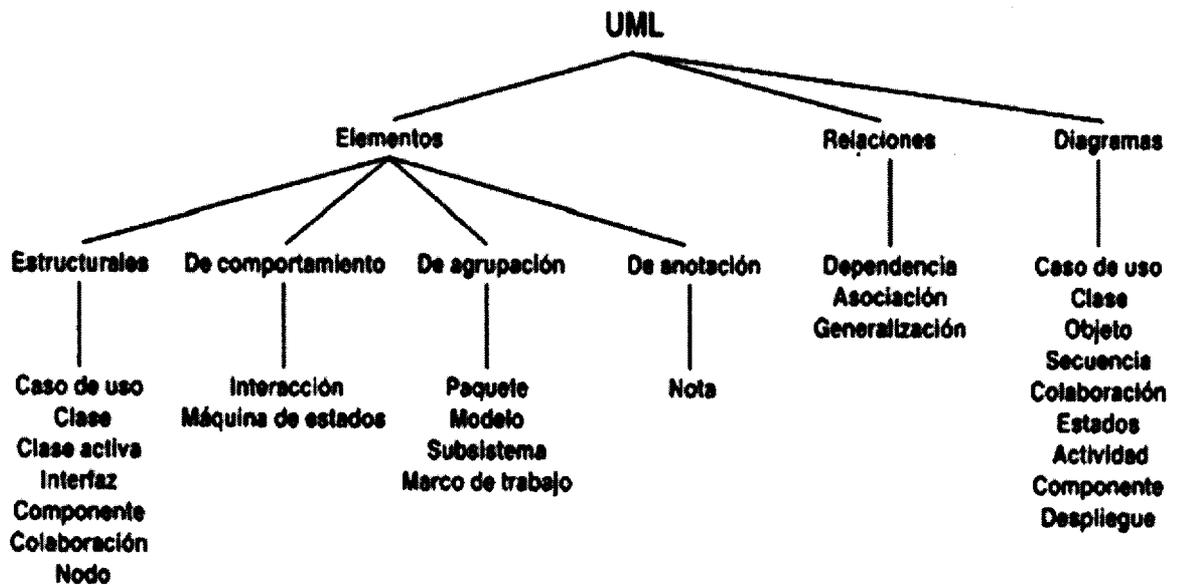


Figura 3: El vocabulario de UML(17)

UML no define un proceso concreto que determine las fases de desarrollo de un sistema, las empresas pueden utilizar UML como el lenguaje para definir sus propios procesos y lo único que tendrán en común con otras organizaciones que utilicen UML serán los tipos de diagramas.

UML es un método independiente del proceso. Los procesos de desarrollo deben ser definidos dentro del contexto donde se van a implementar los sistemas.(18)

2.4 Herramientas a utilizar

2.4.1 Herramienta de Modelado. Rational Rose

Rational Rose es un Idioma Modelado Unificado objeto-orientado (UML) la herramienta de plan de software pensó para el planeando visual y construcción del componente de aplicaciones del software empresa-niveladas. En mucho la misma manera que un director teatral bloquea fuera una obra, un diseñador del software usa a Rose Racional para crear visualmente (modelo) el armazón para una aplicación bloqueando fuera las clases con actores (las figuras del palo), use los elementos del caso (los ovalado), objetos (los rectángulos) y mensajes / las relaciones (las flechas) en un diagrama de la

sucesión que usa los símbolos del arrastrar-y-gota. Rose racional documenta el diagrama como él está construyéndose y entonces genera el código en la opción del diseñador de C++, Elemento esencial Visual, Java, Oracle8, CORBA o Idioma de Definición de Datos.

Dos rasgos populares de Rose Racional son su habilidad de proporcionar desarrollo reiterativo e ingeniería del redondo-viaje. Rose racional les permite a diseñadores aprovecharse la de desarrollo reiterativo (a veces llamó el desarrollo evolutivo) porque la nueva aplicación puede crearse en las fases con el rendimiento de una iteración que se vuelve la entrada al próximo. (Esto está en contraste con desarrollo de la cascada de dónde el proyecto entero se completa empieza a terminar ante un usuario consigue probarlo.) Entonces, cuando el diseñador empieza a entender que cómo los componentes actúan recíprocamente y hacen las modificaciones en el plan, Rose Racional puede realizar lo que se llama la "ingeniería del redondo-viaje remontándose y poniendo al día el resto del modelo para asegurar el código permanece consistente.

Rose racional es extensible, con los complementos del downloadable y las aplicaciones del compañero terceristas. Apoya COM/DCOM (ActiveX), JavaBeans, y Corba componente normas.(19)

Comparación de algunas ediciones:

- Rose Data Modeler: Acelera el plan de la base de datos proporcionando un ambiente modelado visual sofisticado.
- Rose Developer para Java: El Desarrollo modelo-manejado para Java y J2EE.
- Rose Developer para UNIX: Proporciona la herramienta de desarrollo modelo-manejada industria-llevando.
- Rose Developer para Visual Studio: El Desarrollo modelo-manejado para Microsoft VisualStudio 6.
- Rose Enterprise: Un planeando visual y herramienta de desarrollo que usan el Idioma Modelado Unificado.
- Rose Modeler: La herramienta modelada para analistas y arquitectos del software.
- Rose Technical Developer: Una solución de desarrollo modelo-manejada que informatiza la traducción del plan-a-código para Java, C y C++.(20)

2.4.2 Herramientas de desarrollo

2.4.2.1 Lenguajes de Programación

PHP es un lenguaje interpretado de alto nivel embebido en páginas HTML pero sus instrucciones son ejecutadas del lado del servidor. Como producto de código abierto PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y se reparan rápidamente.

El código se pone al día continuamente con mejoras y extensiones del lenguaje para ampliar las capacidades de PHP. Una de sus características más potentes es su soporte para un gran número de gestores de bases de datos.

Ventajas de PHP:

- Muy sencillo de aprender.
- Similar en gran parte de sus sintaxis a C y a PERL.
- Soporta en cierta medida la orientación a objeto. Clases y herencia.
- El análisis léxico para recoger las variables que se pasan en la dirección lo hace PHP de forma automática. Librándose el usuario de tener que separar las variables y sus valores.
- Se puede incrustar código PHP con etiquetas HTML y este código se ejecutará del lado del servidor.
- Soporte de acceso para un gran número de bases de datos.
- Además PHP viene equipado con un conjunto de funciones de seguridad que previenen la interacción de órdenes dentro de una solicitud de datos.
- Permite transmitir por vía HTTP, todo lo que se puede construir con él.

Desventajas de PHP:

- Todo el trabajo lo realiza el servidor y no delega al cliente. Por tanto puede ser más ineficiente a medida que las solicitudes aumenten de número.
- La legibilidad del código puede verse afectada al mezclar sentencias HTML y PHP.
- La orientación a objetos es aún muy deficiente para aplicaciones grandes.

2.4.2.2 PHP5

Con las primeras 2 versiones de PHP, PHP 3 y PHP 4, se había conseguido una plataforma potente y estable para la programación de páginas del lado del servidor. Sin embargo, todavía existían puntos negros en el desarrollo PHP que se han tratado de solucionar con la versión 5, aspectos que se echaron en falta en la versión 4, casi desde el día de su lanzamiento. Se refiere principalmente a la programación orientada a objetos (POO) que, a pesar de que estaba soportada a partir de PHP3, sólo implementaba una parte muy pequeña de las características de este tipo de programación.(21)

PHP5 se apoya en la llamada Zend Engine 2, la nueva versión del motor Zend, que es el corazón de PHP desde la versión 4. Zend 2 supone un auténtico "cambio de paradigma" en la programación con PHP. La versión más reciente de PHP es la 5.2.5 que incluye todas las ventajas que provee el nuevo Zend Engine 2 como:

- Mejor soporte para la Programación Orientada a Objetos, que en versiones anteriores era extremadamente rudimentario, con PHP Data Objects.
- Mejoras de rendimiento.
- Mejor soporte para MySQL con extensión completamente reescrita.
- Mejor soporte a XML (XPath, DOM, etc.).
- Soporte integrado para SOAP.
- Iteradores de datos.
- Manejo de excepciones.

El principal objetivo de PHP5 ha sido mejorar los mecanismos de POO para solucionar las carencias de las anteriores versiones. Un paso necesario para conseguir que PHP sea un lenguaje apto para todo tipo de aplicaciones y entornos, incluso los más exigentes.(21)

Algunas características del trabajo con POO en PHP 5:

- Nombres fijos para los constructores y destructores: hay que utilizar unos nombres predefinidos para los métodos constructores y destructores (Los que se encargan de resumir las tareas de inicialización y destrucción de los objetos. Ahora se han de llamar `__construct()` y `__destruct()`).

- Acceso public, private y protected a propiedades y métodos: se utilizan los modificadores de acceso habituales de la POO. Estos modificadores sirven para definir qué métodos y propiedades de las clases son accesibles desde cada entorno.
- Posibilidad de uso de interfaces: Las interfaces se utilizan en la POO para definir un conjunto de métodos que implementa una clase. Una clase puede implementar varias interfaces o conjuntos de métodos. En la práctica, el uso de interfaces es utilizado muy a menudo para suplir la falta de herencia múltiple de lenguajes como PHP o Java.
- Métodos y clases final: se puede indicar que un método es "final". Con ello no se permite sobrescribir ese método, en una nueva clase que lo herede. Si la clase es "final", lo que se indica es que esa clase no permite ser heredada por otra clase.
- Operador instanceof: se utiliza para saber si un objeto es una instancia de una clase determinada.
- Atributos y métodos static: se hace uso de atributos y métodos static. Son las propiedades y funcionalidades a las que se puede acceder a partir del nombre de clase, sin necesidad de haber instanciado un objeto de dicha clase.
- Clases y métodos abstractos: es posible crear clases y métodos abstractos. Las clases abstractas no se pueden instanciar, se suelen utilizar para heredarlas desde otras clases que no tienen porque ser abstractas. Los métodos abstractos no se pueden llamar, se utilizan más bien para ser heredados por otras clases, donde no tienen porque ser declarados abstractos.
- Constantes de clase: se pueden definir constantes dentro de la clase. Luego se pueden acceder dichas constantes a través de la propia clase.
- Funciones que especifican la clase que reciben por parámetro: se pueden definir funciones y declarar que deben recibir un tipo específico de objeto. En caso que el objeto no sea de la clase correcta, se produce un error.
- Función __autoload(): la función __autoload() sirve para intentar incluir el código de una clase que se necesite, y que no haya sido declarada todavía en el código que se está ejecutando.
- Clonado de objetos: Si se desea, se puede realizar un objeto a partir de la copia exacta de otro objeto. Para ello se utiliza la instrucción "clone". También se puede definir el método __clone() para realizar tareas asociadas con la clonación de un objeto.(4)

2.4.2.3 PHP vs ASP:

Después de hacer un análisis entre el PHP y el ASP, se decide utilizar el PHP debido a que está soportado en la mayoría de las plataformas de Sistemas Operativos, no ocurre lo mismo con ASP por ser propiedad de Microsoft. El lenguaje PHP no tiene costo oculto, o sea que cuando se adquiere incluye un sinnúmero de bibliotecas que proporcionan el soporte para la mayoría de las aplicaciones Web, En el caso de ASP forma parte del Internet Information Server que viene integrado en Windows NT-2000 Server con su elevado costo de adquisición. PHP y ASP son parecidos en cuanto a la forma de utilización, pero PHP es más rápido, gratuito y multiplataforma.

PHP está en el entorno Open Source. Esto quiere decir que en su desarrollo hay un proceso de colaboración que hace que tengas inmediatamente disponibles, de forma gratuita, una enorme cantidad de recursos: el lenguaje en si, el servidor para ejecutarlo, manuales y tutoriales sin fin, y sobre todo scripts, scripts y más scripts que puedes descargar y usar en cuestión de minutos. Esta abundancia de código libremente disponible ayuda aun más en el proceso de aprendizaje.

Uno de los aspectos a tener en cuenta a la hora de seleccionar PHP es que es libre, lo que implica menores costos y servidores más baratos que otras alternativas, a la vez que el tiempo entre el hallazgo de un fallo y su resolución es más corto. Además, el volumen de código PHP libre es mucho mayor que en otras tecnologías. PHP es multiplataforma. Funciona en toda máquina que sea capaz de compilar su código, entre ellas diversos sistemas operativos. El código escrito en PHP en cualquier plataforma funciona exactamente igual en otra. El acceso a las bases de datos de PHP es muy heterogéneo, pues dispone de un juego de funciones distinto por cada gestor. PHP es suficientemente versátil y potente como para hacer tantas aplicaciones grandes que necesiten acceder a recursos a bajo nivel del sistema como pequeños scripts que envíen por correo electrónico un formulario relleno por el usuario. Puede interactuar con muchos motores de bases de datos tales como MySQL, Oracle, PostgreSQL, y otros muchos.

2.4.3 IDE. Eclipse

Eclipse es una comunidad de código abierto cuyos proyectos se centran en la construcción de una plataforma de desarrollo extensible, runtimes y aplicación de marcos para la construcción, despliegue y la gestión de software en todo el ciclo de vida del software.

Cuenta con más de 60 proyectos de código abierto, los cuales pueden ser conceptualmente organizados en siete diferentes pilares o categorías:

1. Desarrollo de la Empresa
2. Embedded y desarrollo de dispositivos
3. Rich Client Platform
4. Rich Internet Applications
5. Aplicación Marcos
6. Application Lifecycle Management (ALM)
7. Arquitectura Orientada a Servicios (SOA)

También cuenta con el apoyo de un gran y vibrante ecosistema de importantes proveedores de soluciones de TI, las nuevas empresas innovadoras, universidades e instituciones de investigación y los particulares que extender, apoyar y complementar la Plataforma Eclipse.

Eclipse utiliza el Eclipse Public License (EPL). El EPL es una licencia comercial amigable que permite a las organizaciones de software Eclipse incluir en sus productos comerciales, al mismo tiempo pedir a los que crear obras derivadas del EPL código de contribuir a la comunidad.(22)

La versión actual de Eclipse dispone de las siguientes características:

- Editor de texto
- Resaltado de sintaxis
- Compilación en tiempo real
- Pruebas unitarias con JUnit
- Control de versiones con CVS
- Integración con Ant
- Asistentes (wizards): para creación de proyectos, clases, tests, etc.
- Refactorización

Así mismo, a través de plugins libremente disponibles es posible añadir:

- Control de versiones con Subversión.
- Integración con Hibernate.(23)

2.5 Gestor de Base de Datos

Un sistema gestor de base de datos se define como el conjunto de programas que administran y gestionan la información contenida en una base de datos. Ayuda a realizar las siguientes acciones:

- Definición de los datos
- Mantenimiento de la integridad de los datos dentro de la base de datos
- Control de la seguridad y privacidad de los datos
- Manipulación de los datos

Un sistema gestor de base de datos está compuesto de:

1. El gestor de la base de datos

Se trata de un conjunto de programas no visibles al usuario final que se encargan de la privacidad, la integridad, la seguridad de los datos y la interacción con el sistema operativo. Proporciona una interfaz entre los datos, los programas que los manejan y los usuarios finales.

Cualquier operación que el usuario hace contra la base de datos está controlada por el gestor.

El gestor almacena una descripción de datos en lo que llamamos diccionario de datos, así como los usuarios permitidos y los permisos.

Tiene que haber un usuario administrador encargado de centralizar todas estas tareas.

2. Diccionario de datos

Es una base de datos donde se guardan todas las propiedades de la base de datos, descripción de la estructura, relaciones entre los datos, etc.

El diccionario debe contener:

- La descripción externa, conceptual e interna de la base de datos
- Las restricciones sobre los datos
- El acceso a los datos
- Las descripciones de las cuentas de usuario

- Los permisos de los usuarios
- Los esquemas externos de cada programa

3. El administrador de la base de datos

Es una persona o grupo de personas responsables del control del sistema gestor de base de datos.

Las principales tareas de un administrador son:

- La definición del esquema lógico y físico de la base de datos
- La definición de las vistas de usuario
- La asignación y edición de permisos para los usuarios
- Mantenimiento y seguimiento de la seguridad en la base de datos
- Mantenimiento general del sistema gestor de base de datos

4. Los lenguajes

Un sistema gestor de base de datos debe proporcionar una serie de lenguajes para la definición y manipulación de la base de datos. Estos lenguajes son los siguientes:

- Lenguaje de definición de datos (DDL). Para definir los esquemas de la base de datos
- Lenguaje de manipulación de datos (DML). Para manipular los datos de la base de datos
- Lenguaje de control de datos (DCL). Para la administración de usuarios y seguridad en la base de datos.(24)

2.5.1 MySQL

MySQL, es el sistema de gestión de bases de datos SQL más popular. Lo desarrolla, distribuye y soporta MySQL AB. MySQL AB es una compañía comercial, fundada por los desarrolladores de MySQL.

MySQL es un sistema de gestión de bases de datos

Una base de datos es una colección estructurada de datos. Puede ser cualquier cosa, desde una simple lista de compra a una galería de pintura o las más vastas cantidades de información en una red

corporativa. Para añadir, acceder, y procesar los datos almacenados en una base de datos, necesita un sistema de gestión de base de datos como MySQL Server.

MySQL es un sistema de gestión de bases de datos relacionales

Una base de datos relacional almacena datos en tablas separadas en lugar de poner todos los datos en un gran almacén. Esto añade velocidad y flexibilidad. La parte SQL de MySQL se refiere a Structured Query Language. SQL es el lenguaje estandarizado más común para acceder a bases de datos y está definido por el estándar ANSI/ISO SQL. El estándar SQL ha evolucionado desde 1986 y existen varias versiones. En este manual, "SQL-92" se refiere al estándar del 1992, "SQL: 1999" se refiere a la versión del 1999, y "SQL: 2003" se refiere a la versión actual del estándar. Se usa la frase "el estándar SQL" para referirnos a la versión actual de SQL.

MySQL software es Open Source; Open Source significa que es posible para cualquiera usar y modificar el software. Cualquiera puede bajar el software MySQL desde internet y usarlo sin pagar nada. Si lo desea, puede estudiar el código fuente y cambiarlo para adaptarlo a sus necesidades. El software MySQL usa la licencia GPL (GNU General Public License), para definir lo que puede y no puede hacer con el software en diferentes situaciones. Si no se encuentra cómodo con la GPL o necesita añadir código MySQL en una aplicación comercial, puede comprarnos una licencia comercial. Consulte la Introducción a las Licencias MySQL para más información.

El servidor de base de datos MySQL es muy rápido, fiable y fácil de usar. El servidor MySQL también tiene una serie de características prácticas desarrolladas en cooperación con los usuarios.

MySQL Server se desarrolló originalmente para tratar grandes bases de datos mucho más rápido que soluciones existentes y ha sido usado con éxito en entornos de producción de alto rendimiento durante varios años. MySQL Server ofrece hoy en día una gran cantidad de funciones. Su conectividad, velocidad, y seguridad hacen de MySQL Server altamente apropiado para acceder bases de datos en Internet.

MySQL Server trabaja en entornos cliente/servidor o incrustados, consiste en un servidor SQL multi-threaded que trabaja con diferentes bakends, programas y bibliotecas cliente, herramientas administrativas y un amplio abanico de interfaces de programación para aplicaciones (APIs).

También se proporciona el MySQL Server como biblioteca incrustada multi-threaded que puede linkar en su aplicación para obtener un producto más pequeño, rápido y fácil de administrar.

Una gran cantidad de software de contribuciones está disponible para MySQL. Es muy posible que su aplicación o lenguaje favorito soporte el servidor de base de datos MySQL. (25)

La siguiente lista describe algunas de las características más importantes del software de base de datos MySQL:

- Interioridades y portabilidad.
- Funciona en diferentes plataformas. Usa GNU Automake, Autoconf, y Libtool para portabilidad.
- APIs disponibles para C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, y Tcl.
- Usa tablas en disco B-tree (MyISAM) muy rápidas con compresión de índice.
- Relativamente sencillo de añadir otro sistema de almacenamiento. Esto es útil si desea añadir una interfaz SQL para una base de datos propia
- Un sistema de reserva de memoria muy rápido basado en threads.
- Tablas hash en memoria, que son usadas como tablas temporales.
- El código MySQL se prueba con Purify (un detector de memoria perdida comercial) así como con Valgrind, una herramienta GPL

Tipos de columnas

- Diversos tipos de columnas: enteros con/sin signo de 1, 2, 3, 4, y 8 bytes de longitud, FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET, ENUM, y tipos espaciales OpenGIS.
- Registros de longitud fija y longitud variable.

Sentencias y funciones

- Soporte completo para operadores y funciones en las cláusulas de consultas SELECT y WHERE
- Soporte completo para las cláusulas SQL GROUP BY y ORDER BY. Soporte de funciones de agrupación (COUNT(), COUNT (DISTINCT), AVG(), STD(), SUM(), MAX(), MIN(), y GROUP_CONCAT()).
- Soporte para LEFT OUTER JOIN y RIGHT OUTER JOIN cumpliendo estándares de sintaxis SQL y ODBC.

- Soporte para alias en tablas y columnas como lo requiere el estándar SQL.
- DELETE, INSERT, REPLACE, y UPDATE devuelven el número de filas que han cambiado (han sido afectadas). Es posible devolver el número de filas que serían afectadas usando un flag al conectar con el servidor.

Seguridad

Un sistema de privilegios y contraseñas que es muy flexible y seguro, y que permite verificación basada en el host. Las contraseñas son seguras porque todo el tráfico de contraseñas está encriptado cuando se conecta con un servidor.

Escalabilidad y límites

- Soporte a grandes bases de datos. Se utiliza MySQL Server con bases de datos que contienen 50 millones de registros.
- Se permiten hasta 64 índices por tabla (32 antes de MySQL 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de MySQL 4.1.2). Un índice puede usar prefijos de una columna para los tipos de columna CHAR, VARCHAR, BLOB, o TEXT.

Conectividad

- Los clientes pueden conectar con el servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows de la familia NT (NT, 2000, XP, o 2003), los clientes pueden usar named pipes para la conexión. En sistemas Unix, los clientes pueden conectar usando ficheros socket Unix.
- En MySQL 5.0, los servidores Windows soportan conexiones con memoria compartida si se inicializan con la opción --shared-memory. Los clientes pueden conectar a través de memoria compartida usando la opción --protocol=memory.

Localización

- El servidor puede proporcionar mensajes de error a los clientes en muchos idiomas.
- Soporte completo para distintos conjuntos de caracteres, incluyendo latin1 (ISO-8859-1), german, big5 y más. Por ejemplo, los caracteres escandinavos 'â', 'ä' y 'ö' están permitidos en nombres de tablas y columnas. El soporte para Unicode está disponible.

- Todos los datos se guardan en el conjunto de caracteres elegido. Todas las comparaciones para columnas normales de cadenas de caracteres son case-insensitive.

Clientes y herramientas

MySQL server tiene soporte para comandos SQL para chequear, optimizar, y reparar tablas. Estos comandos están disponibles a través de la línea de comandos y el cliente mysqlcheck. MySQL también incluye myisamchk, una utilidad de línea de comandos muy rápida para efectuar estas operaciones en tablas MyISAM. Todos los programas MySQL pueden invocarse con las opciones --help o -? para obtener asistencia en línea.(26)

MySQL Comparado con otras Bases de Datos:

El Gestor de bases de datos, MySQL, comparado con los de su categoría, es considerado como el más rápido y robusto tanto para volúmenes de datos grandes como pequeños. Pero esta rapidez es a costa de no implementar ciertos aspectos del SQL.

Estos aspectos son por un lado los Triggers y por otro lado la Lógica Transaccional.

Los Triggers no es nada más que una porción de código almacenado que se "dispara" o se ejecuta cuando se realiza una operación (actualización, borrado, etc.) con la base de datos. Naturalmente comprobar la propia existencia de disparador y ejecutarlo si existe consume recursos y tiempo y es como su propio manual indica, la única razón por la que los Triggers no están soportados.

Un aspecto muy importante en cualquier base de datos relacional es la consistencia de las diferentes tablas que la componen, para conseguir esto de una forma más o menos fácil es utilizando la "Lógica Transaccional", será el propio gestor de base de datos el que proporcione mecanismos de bloqueo de ficheros y consolidación o retroceso en la operaciones con las tablas. Pues bien MySQL no soporta las transacciones en aras simplemente de la velocidad (o por lo menos así lo indican en sus comentarios), sólo es posible apoyarse con los comandos LOCK tables /UNLOCK tables que permiten bloquear tablas impidiendo que otros usuarios puedan acceder a ellas pero sin la posibilidad de deshacer las operaciones realizadas con los datos.

MySQL comparado con MSQL:

MSQL deber ser más rápido al:

- Hacer pruebas que realizan repetidas conexiones y desconexiones, ejecutando una consulta muy simple durante cada conexión.
- Operaciones INSERT muy simples en las tablas, con pocas columnas y llaves.
- CREATE TABLE y DROP TABLE.
- SELECT sobre algo que no sea un índice. (examinar una tabla es muy fácil.)

Después de que la conexión se establece, MySQL debe realizarlo mucho mejor. Por otro lado, MySQL es mucho más rápido que MSQL en la mayoría de las otras implementaciones del SQL. Por ejemplo en lo siguiente:

- Operaciones SELECT complejas.
- Recuperando resultados grandes (MySQL tiene un protocolo mejor, más rápido, y más seguro).
- Tablas con strings de longitud variable, porque MySQL es más eficiente al manipularlos y puede tener índices en columnas de VARCHAR.
- Manejando tablas con muchas columnas.
- Manejando tablas con grandes longitudes de registro.
- SELECT con muchas expresiones.
- SELECT sobre grandes tablas.
- Manejando muchas conexiones al mismo tiempo. MySQL se multi-enhebra totalmente. Cada conexión tiene su propio hilo de modo que ningún hilo tiene que esperar por otro (a menos que un hilo esté modificando una tabla y otro hilo quiera acceder). Una vez que en MSQL, una conexión se establece, todos los otros deben esperar hasta que el primero ha terminado, sin tener en cuenta si la conexión está ejecutando una consulta que es corta o que requiere mucho tiempo. Cuando la primera conexión termina, el próximo puede servirse, mientras todos los otros esperan de nuevo, etc.
- Joins. MSQL puede ponerse lento patológicamente si se cambia el orden de tablas en un SELECT. En tiempo más de 15000 veces más lento que MySQL. Esto es debido a la falta de un optimizador en MSQL para pedir tablas en el orden óptimo. ¡Sin embargo, si se pusieran las

tablas en exactamente el orden correcto en mSQL2 y el WHERE es simple y usa columnas con índice, el Join será relativamente rápido.

- ORDER BY y GROUP BY.
- DISTINCT.
- Usando TEXT o BLOB columnas.

MySQL vs PostgreSQL:

Hay muchos comentarios sobre MySQL y PostgreSQL para ver cuál puede ser mejor, y la información es abundante pero confusa, la mayoría de ellos catalogables dentro de las "guerras santas" como las que se producen en las comparaciones "Linux vs. Windows", "Mac vs. PC", etc. De entre todo esto se concluye:

MySQL:

Su principal objetivo de diseño fue la velocidad. Se sacrificaron algunas características esenciales en sistemas más serios con este fin. Otra característica importante es que consume muy pocos recursos, tanto de CPU como de memoria. Licencia GPL a partir de la versión 3.23.19.

Ventajas:

- Mayor rendimiento. Mayor velocidad tanto al conectar con el servidor como al servir selects y demás.
- Mejores utilidades de administración (backup, recuperación de errores, etc.).
- Aunque se cuelgue, no suele perder información ni corromper los datos.
- Mejor integración con PHP.
- No hay límites en el tamaño de los registros.
- Mejor control de acceso, en el sentido de qué usuarios tienen acceso a qué tablas y con qué permisos.
- MySQL se comporta mejor que PostgreSQL a la hora de modificar o añadir campos a una tabla "en caliente".

Inconvenientes:

- No soporta transacciones, "roll-backs" ni subselects.
- No considera las claves ajenas.

- Ignora la integridad referencial, dejándola en manos del programador de la aplicación.

PostgreSQL:

PostgreSQL intenta ser un sistema de bases de datos de mayor nivel que MySQL, a la altura de Oracle, Sybase o Interbase. Licencia BSD.

Ventajas:

- Por su arquitectura de diseño, escala muy bien al aumentar el número de CPUs y la cantidad de RAM.
- Soporta transacciones y desde la versión 7.0, claves ajenas (con comprobaciones de integridad referencial).
- Tiene mejor soporte para Triggers y procedimientos en el servidor.
- Soporta un subconjunto de SQL92 MAYOR que el que soporta MySQL. Además, tiene ciertas características orientadas a objetos.

Inconvenientes:

- Consume BASTANTES más recursos y carga más el sistema.
- Límite del tamaño de cada fila de las tablas a 8k!!! (se puede ampliar a 32k recompilando, pero con un coste añadido en el rendimiento).
- Es de 2 a 3 veces más lenta que MySQL.
- Menos funciones en PHP.
- En cuanto a consideraciones de estabilidad del servidor, cada comparativa da datos contradictorios. En general parece que MySQL es más estable (aunque también hay personas que opina lo contrario), y que PostgreSQL tiende a desperdiciar memoria y sobrecargar bastante el sistema (aunque de nuevo, hay opiniones distintas).

Como conclusión a la comparación entre MySQL y PostgreSQL, parece aceptado que MySQL junto con Apache y PHP forman un buen equipo para servir páginas web con contenido dinámico, discusiones, noticias, etc., por ejemplo al estilo de SlashDot. En general, sistemas en los que la velocidad y el número de accesos concurrentes sea algo primordial, y la seguridad no sea muy importante (pueda bastar con hacer backups periódicos que se restaurarán tras una caída del servidor). En cambio, para sistemas más serios en las que la consistencia de la BD sea fundamental

(BD con información realmente importante, bancos, etc.) PostgreSQL es una mejor opción pese a su mayor lentitud.(27)

2.6 Conclusiones parciales

En este capítulo se realizó el estudio de algunas herramientas utilizadas para el desarrollo de aplicaciones web. Para el análisis y diseño, se utilizó metodología RUP, MySQL como gestor de base de datos que constituye una aplicación poderosa, robusta, que permite gran seguridad, y tiene múltiples motores de almacenamiento. Además, se hace uso de PHP como lenguaje de programación y más específicamente PHP 5 ya que es el más conveniente para el trabajo con el Eclipse que fue el IDE seleccionado por las características que presenta.

Capítulo 3: Presentación de la solución propuesta

3.1 Introducción

En este capítulo se realiza el análisis del proceso de negocio correspondiente al objeto de estudio y se comienza con la modelación del sistema propuesto. Se realiza primeramente todo el análisis del negocio del problema para posteriormente realizar el proceso de la captura de requerimientos del sistema, tanto los funcionales como no funcionales. Se realiza la construcción de los casos de uso del sistema y a los diagramas que representan a la solución propuesta del sistema y seguidamente, la descripción textual de cada uno de estos casos de uso.

3.2 Modelo de Negocio

Para realizar el modelo de negocio, primeramente se seleccionan los actores y trabajadores que intervienen en el negocio, luego se definen los casos de usos que lo integran conformando el diagrama de caso de uso del negocio. Por último con los trabajadores y los objetos utilizados en el proceso se confecciona el modelo de objetos del negocio, cumpliendo así con la salida de todos los artefactos del flujo de trabajo.

3.2.1 Actores y trabajadores del negocio

Un actor del negocio es cualquier individuo, grupo, organización que interactúa con el negocio. Siempre permanece fuera de las fronteras del negocio que se investiga.

Actores del negocio	Justificación
Usuario	Es el actor que se beneficia con la realización de autoaprendizaje de idiomas, ya que es el principal beneficiado con el resultado de las diferentes actividades que se propuso realizar. El usuario puede ser cualquier estudiante, profesor o trabajador UCI que visite el CASIE.

Tabla 1: Justificación de los actores del negocio

Trabajador del negocio representa a personas o sistemas (software) dentro del negocio que son las que realizan las actividades que están comprendidas dentro de un caso de uso.

Trabajadores del negocio	Justificación
Recepcionista	Se encarga de registrar todos los datos de los clientes que visiten los CASIE, así como crear los contratos cuando los clientes solicitan el Autoaprendizaje.
Asesor	Se encarga de emitir una calificación de los Test de nivel aplicado a los clientes, así como orientarlos con las posibles rutas a seguir para su Autoaprendizaje.
Técnico General	Se encarga de aplicar los Test de nivel a los clientes una vez creado el contrato, así como registrar los datos de cuando se utilice un material para el Autoaprendizaje.
Técnico Informático	Se encarga de instalar el software que los clientes necesiten para su Autoaprendizaje.

Tabla 2: Justificación de los trabajadores del negocio

3.2.2 Procesos de negocio

En el negocio del Autoaprendizaje de las lenguas extranjeras se realizan 4 procesos fundamentales: la recepción, asesoría, aplicación de los Test de nivel e instalación del software que los clientes necesiten (estos procesos se realizan en el CASIE).

En la recepción, la recepcionista se encarga de registrar todos los datos del cliente una vez llegado este al CASIE y crea además los contratos cuando estos solicitan el Autoaprendizaje.

En la asesoría al cliente, el asesor emite una calificación de los Test de nivel aplicado a estos y los orienta con las posibles rutas a seguir para su Autoaprendizaje.

En la aplicación de los Test de nivel, el Técnico General aplica este a los clientes una vez creado el contrato y registra los datos de cuando se utilice un material para el Autoaprendizaje.

En caso de que un cliente necesite algún software que no esté instalado para el desarrollo de su Autoaprendizaje, el Técnico Informático es el encargado de realizar dicha instalación.

3.2.3 Diagrama de Casos de Uso del Negocio

Un diagrama de Casos de Uso del Negocio representa un conjunto de casos de uso, actores y las relaciones entre ellos.

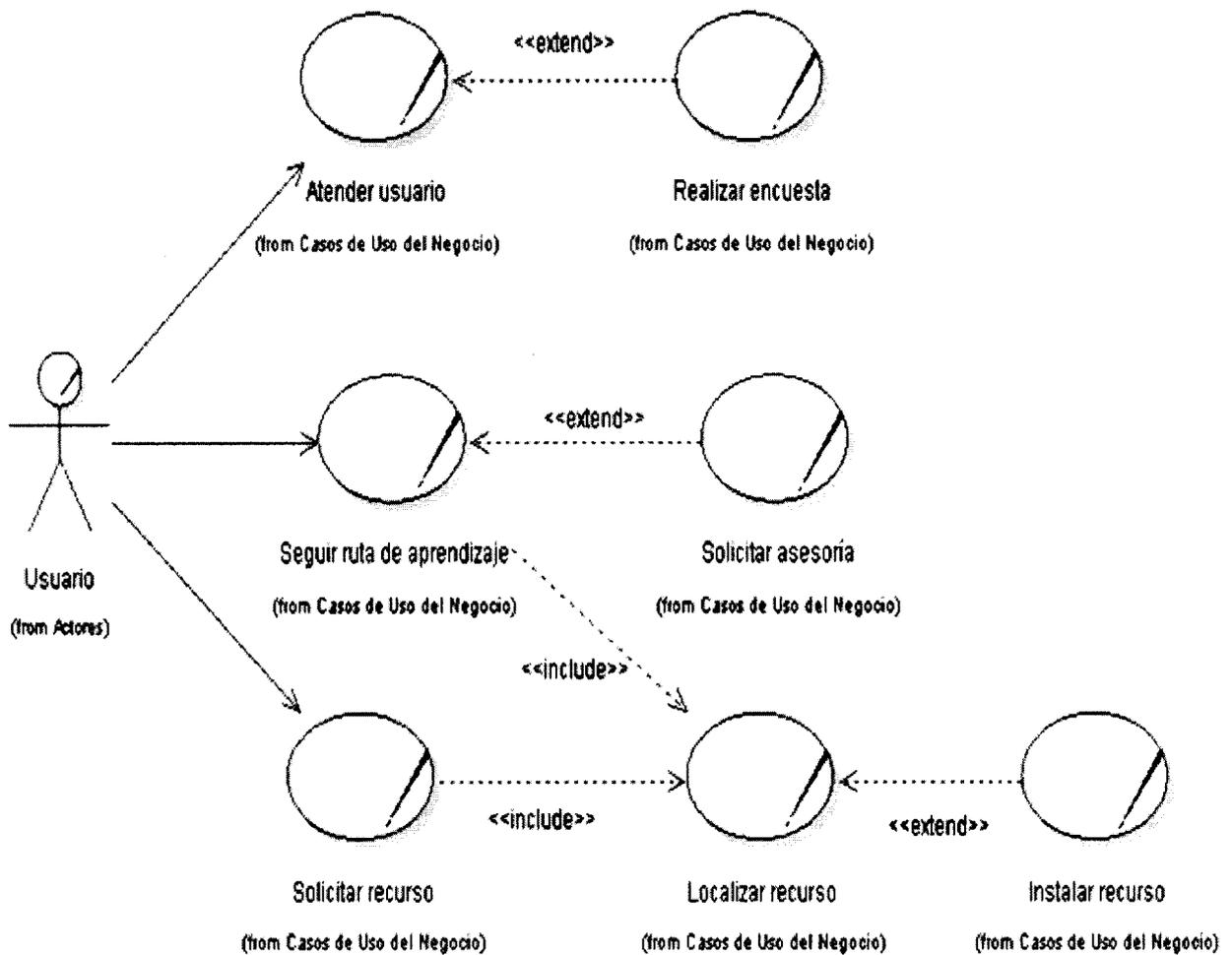


Figura 4: Diagrama de Casos de Uso del Negocio

3.2.4 Descripción textual de los Casos de Uso de Negocio

Caso de Uso:	Atender usuario
Actores:	Usuario(inicia)
Trabajadores:	Recepcionista, técnico general
Resumen:	El caso de uso se inicia cuando un usuario llega al CASIE y solicita el servicio de este para el Autoaprendizaje.
Precondiciones:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1- El caso de uso se inicia cuando el usuario llega al CASIE y solicita Auto-aprender un idioma.	2- La recepcionista le da la bienvenida al CASIE y le pide los datos del usuario.
3- El usuario informa los datos.	4- La recepcionista recibe la información del usuario. 5- La recepcionista registra los datos en el registro de clientes. 6- La recepcionista verifica si es la primera vez que el usuario viene al CASIE. 7- Si es la primera vez, la recepcionista le informa al técnico general que le aplique una encuesta al usuario finalizando así el caso de uso, sino ver Flujo Alterno 1.
Flujos Alternos 1	
Acción del Actor	Respuesta del Negocio
	1- Sino es la primera vez que el usuario viene al CASIE la recepcionista consulta la fase y nivel del usuario en el libro de categorías. 2- La recepcionista informa al usuario fase y nivel en que se encuentra.

3- El usuario recibe la información, terminando así el caso de uso.	
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
Poscondiciones	El usuario queda registrado en el registro de clientes.
Prioridad	Crítico

Tabla 3: Descripción Textual del Casos de Uso Atender Usuario

Caso de Uso:	Realizar Encuesta.
Actores:	Usuario
Trabajadores:	Técnico general(inicia), Asesor
Resumen:	El caso de uso se inicia cuando el técnico general entrega la encuesta al usuario, el usuario realiza la encuesta, el asesor califica la encuesta e informa la fase, nivel y la ruta a seguir.
Precondiciones:	Realizar Autoaprendizaje por primera vez en el CASIE.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
	1- El técnico general recibe orientación de aplicar encuesta. 2- El técnico general entrega la encuesta al usuario.
3- El usuario recibe la encuesta. 4- El usuario responde la encuesta. 5- El usuario entrega la encuesta al técnico general.	6- El técnico general recibe la encuesta respondida por el usuario. 7- El técnico general entrega la encuesta respondida por el usuario al asesor.
	8- El asesor recibe la encuesta. 9- El asesor califica la encuesta. 10- El asesor registra la fase y el nivel en

	que se encuentra el usuario en el registro de categorías. 11- El asesor crea un contrato en convenio con el usuario.
12- El usuario firma el contrato. 13- El usuario entrega el contrato al asesor.	14- El asesor recoge el contrato. 15- El asesor firma el contrato. 16- El asesor informa al usuario fase, nivel, la ruta de aprendizaje a seguir, finalizando así el caso de uso.
Flujos Alternos	
Acción del Actor	Respuesta del Negocio
Poscondiciones	Queda confeccionada la ruta de aprendizaje del usuario.
Prioridad	Critico

Tabla 4: Descripción Textual del Casos de Uso Realizar encuesta

Caso de Uso:	Solicitar Asesoría
Actores:	Usuario (inicia)
Trabajadores:	Asesor
Resumen:	El caso de uso se inicia cuando el cliente solicita la ayuda del asesor, debido a la existencia de una duda en la ruta de Autoaprendizaje trazada. El asesor brinda la asesoría al cliente finalizando el caso de uso.
Precondiciones:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1- El usuario solicita la ayuda del asesor.	2- El asesor recibe el pedido de ayuda por parte del usuario. 3- El asesor pregunta al usuario las dudas que este tiene.
4- El usuario le informa la duda que tiene.	5- El asesor recibe la información de la duda

	del usuario. 6- El asesor analiza la pregunta realizada. 7- Si la pregunta puede ser respondida, el asesor emite la respuesta de la pregunta al usuario, sino ir a Flujos Alternos 1.
8- El usuario recibe la respuesta del asesor a la duda planteada finalizando así el caso de uso.	
Flujos Alternos 1	
Acción del Actor	Respuesta del Negocio
	1- Si la duda no puede ser aclarada el asesor comunica al usuario no estar en condiciones para aclarar la duda.
2- El usuario recibe la respuesta del asesor de no estar en condiciones de aclarar la duda planteada.	
2- Si el usuario tiene mas dudas ir a la actividad 4 del flujo normal de eventos, sino ir a Flujos Alterno 2.	
Flujos Alternos 2	
Acción del Actor	Respuesta del Negocio
1- Si el usuario no tiene más dudas abandona al asesor y continúa con su aprendizaje finalizando así el caso de uso	
Poscondiciones	El usuario aclaró o no las dudas existentes en la tarea que estaba realizando.
Prioridad	Crítico

Tabla 5: Descripción Textual del Casos de Uso Solicitar Asesoría

Caso de Uso:	Seguir ruta de aprendizaje
Actores:	Usuario(inicia)
Trabajadores:	Asesor
Resumen:	El caso de uso se inicia cuando el usuario decide seguir una ruta de

	aprendizaje para lo cual cuenta con una lista de actividades y recursos la cual debe seguir.	
Precondiciones:		
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Negocio
	1- El caso de uso se inicia cuando el usuario llega al CASIE para seguir una ruta de aprendizaje 2- El usuario solicita a la recepcionista ver el contrato para comprobar ruta de aprendizaje.	3- La recepcionista busca el contrato del usuario. 4- La recepcionista entrega el contrato al usuario.
	5- El usuario comprueba la ruta de aprendizaje. 6- El usuario escoge la actividad que esta al principio de la lista de actividades no realizadas. 7- El usuario entrega el contrato a la recepcionista.	8- La recepcionista recoge el contrato. 9- La recepcionista guarda el contrato del usuario finalizando así el caso de uso.
Flujos Alternos		
	Acción del Actor	Respuesta del Negocio
Poscondiciones	El usuario escoge las actividades según la ruta de aprendizaje trazada por el asesor.	
Prioridad	Crítico	

Tabla 6: Descripción Textual del Casos de Uso Seguir ruta de aprendizaje

Caso de Uso:	Solicitar recurso.
Actores:	Usuario (inicia)
Trabajadores:	Técnico general.
Resumen:	El caso de uso se inicia cuando un usuario solicita al Técnico General un recurso. El Técnico General recibe la petición y brinda al cliente la localización del recurso solicitado, finalizando el caso de uso.
Precondiciones:	
Flujo Normal de Eventos	

Acción del Actor		Respuesta del Negocio
1- El caso de uso se inicia cuando el usuario llega al CASIE y solicita consultar un recurso.		2- El técnico general recibe la solicitud del usuario. 3- El técnico general informa localización del recurso al usuario.
4- El usuario recibe la información.		5- El técnico general registra los datos de la solicitud en el registro de consulta de libro.
Flujos Alternos		
Acción del Actor		Respuesta del Negocio
Poscondiciones	Al usuario se le entregó un recurso y quedó registrado en el Registro Consulta de Libro.	
Prioridad	Crítico	

Tabla 7: Descripción Textual del Casos de Uso Solicitar Recurso

Caso de Uso:	Localizar recurso.
Actores:	Usuario
Trabajadores:	Técnico general(inicia)
Resumen:	El caso de uso se inicia dada la solicitud de un recurso por un usuario y el técnico general empieza la gestión de localizar el recurso en los CASIE.
Precondiciones:	Solicitar un recurso o solicitar seguir ruta de aprendizaje.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
	1- El técnico general busca en el listado de recursos del CASIE. 2- Si el recurso aparece en el listado de recursos del CASIE el técnico general verifica la localización del recurso sino ver flujo alterno 1. 3- Verifica el tipo del recurso encontrado. 4- Si el recurso no es digital informa al usuario la localización del recurso, sino ver

	flujo alternativo 2.
5- El usuario recibe la información terminando así el caso de uso.	
Flujos Alterno 1	
Acción del Actor	Respuesta del Negocio
	<p>1- Si el recurso no aparece en el listado de recursos del CASIE se busca en los listados de recursos de los demás CASIE.</p> <p>2- Si el recurso solicitado no se encuentra en ningún listado, el técnico general informa al usuario que no disponen del recurso solicitado, sino ver flujo alternativo 3.</p>
3- El usuario recibe la información terminando así el caso de uso.	
Flujos Alterno 2	
Acción del Actor	Respuesta del Negocio
	<p>1- Si el recurso es digital el técnico general verifica si el recurso está instalado.</p> <p>2- Si el recurso está instalado informa al usuario la localización del recurso, sino ver flujo alternativo 4.</p>
3- El usuario recibe la información terminando así el caso de uso.	
Flujos Alterno 3	
Acción del Actor	Respuesta del Negocio
	1- Si el recurso se encuentra en el listado de otro CASIE, ir a la acción 2 del flujo normal de eventos.
Flujos Alterno 4	
Acción del Actor	Respuesta del Negocio

	1- Si el recurso no está instalado comunica al técnico informático instalar el recurso. 2- Informa al usuario la situación.
3- El usuario recibe la información finalizando así el caso de uso.	
Poscondiciones	Al usuario se le localiza el recurso solicitado en el propio CASIE o en los demás CASIE de la UCI.
Prioridad	Crítico

Tabla 8: Descripción Textual del Casos de Uso Localizar Recurso

Caso de Uso:	Instalar recurso.
Actores:	Usuario
Trabajadores:	Técnico Informático(Inicia), Técnico general
Resumen:	El caso de uso se inicia cuando el recurso solicitado por un usuario es digital y no está instalado y el técnico general orienta al técnico informático instalar el recurso.
Precondiciones:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
	<ol style="list-style-type: none"> 1. El técnico informático recibe la orientación del técnico general. 2. El técnico informático comprueba poder instalar el recurso. 3. Si el recurso se puede instalar el técnico informático instala el recurso, sino ver flujo alterno 1. 4. El técnico informático informa la disponibilidad del recurso al técnico general. 5. El técnico general recibe confirmación de la instalación del recurso.

	6. El técnico general informa al usuario que ya el recurso está instalado.
7. El usuario recibe la información de la disponibilidad del recurso solicitado finalizando así el caso de uso.	
Flujos Alterno 1	
Acción del Actor	Respuesta del Negocio
	<p>1. Si el recurso solicitado no se puede instalar el técnico informático informa al técnico general que el recurso solicitado no está disponible.</p> <p>2. El técnico general recibe la información.</p> <p>3. El técnico general informa al usuario.</p>
4. El usuario recibe la información finalizando así el caso de uso.	
Poscondiciones	Al cliente se le entregó un recurso y quedó registrado en el Registro Consulta de Libro y en el Registro del Cliente.
Prioridad	Crítico

Tabla 9: Descripción Textual del Casos de Uso Instalar Recurso

A continuación se muestran los diagramas de actividades de los Casos de Uso anteriormente mencionados. (Ver anexos)

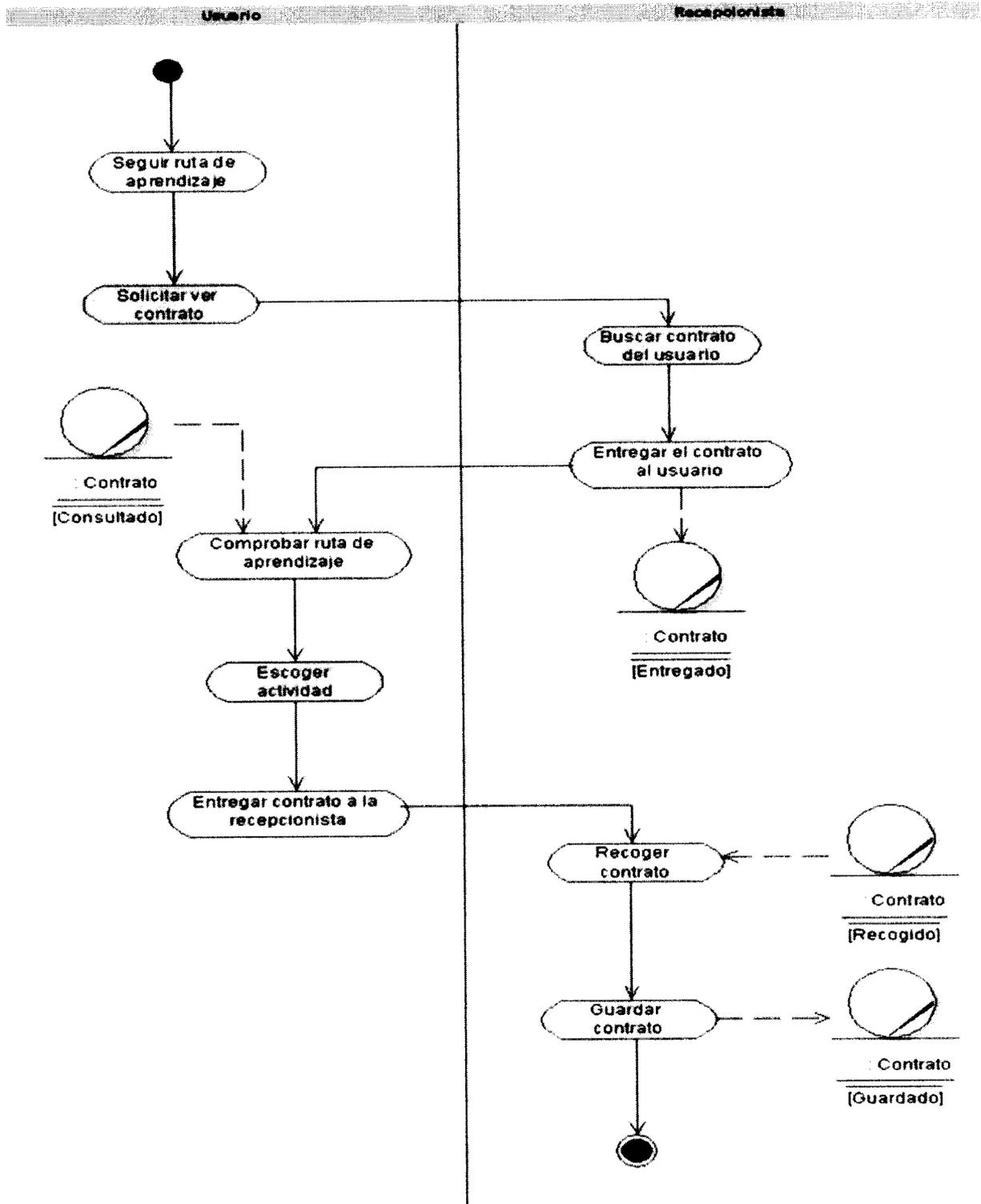


Figura 5: Diagrama de actividades del Caso de Uso Seguir ruta de aprendizaje

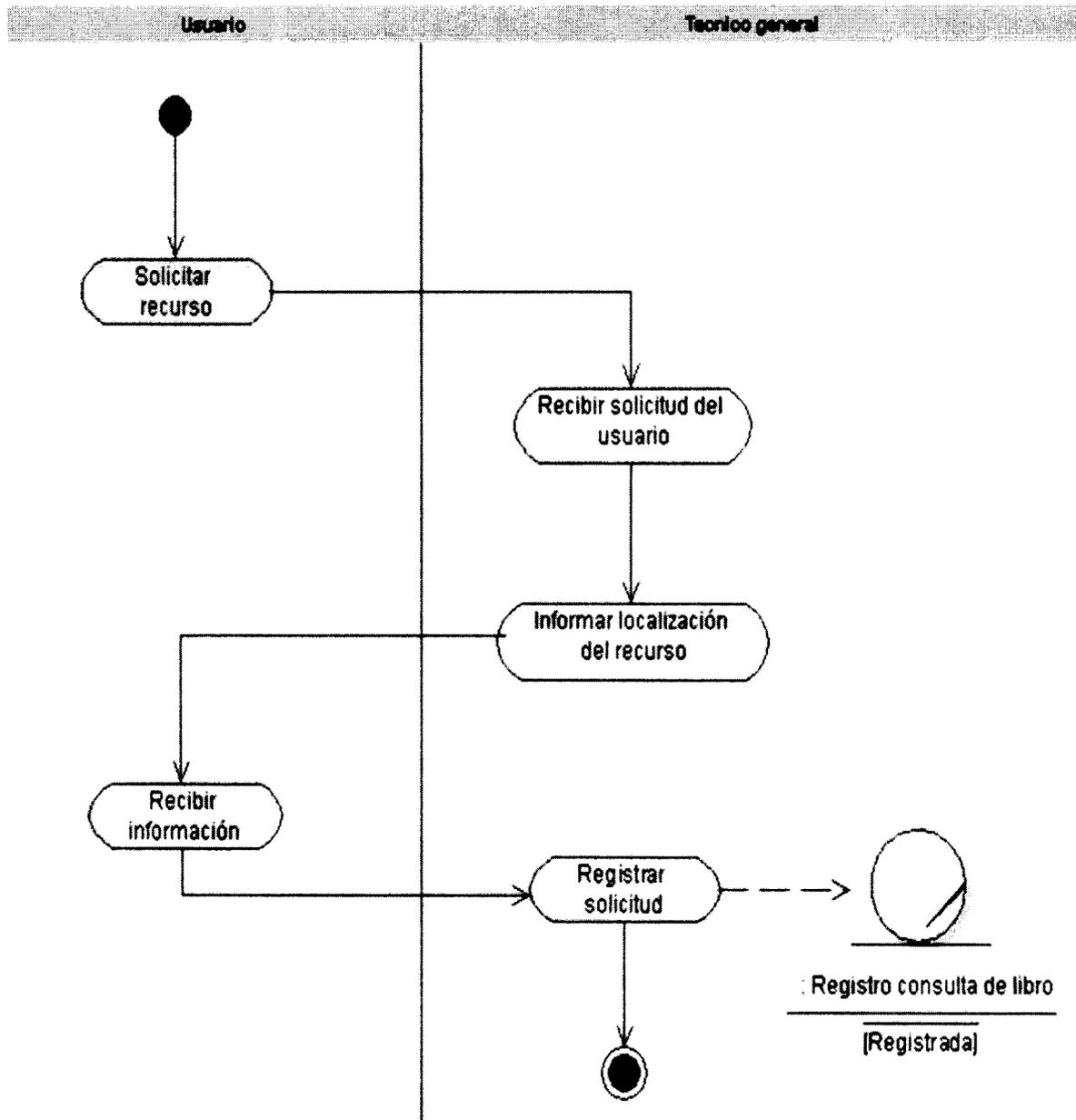


Figura 6: Diagrama de actividades del Casos de Uso Solicitar Recurso

3.2.5 Diagrama de Modelo de Objetos

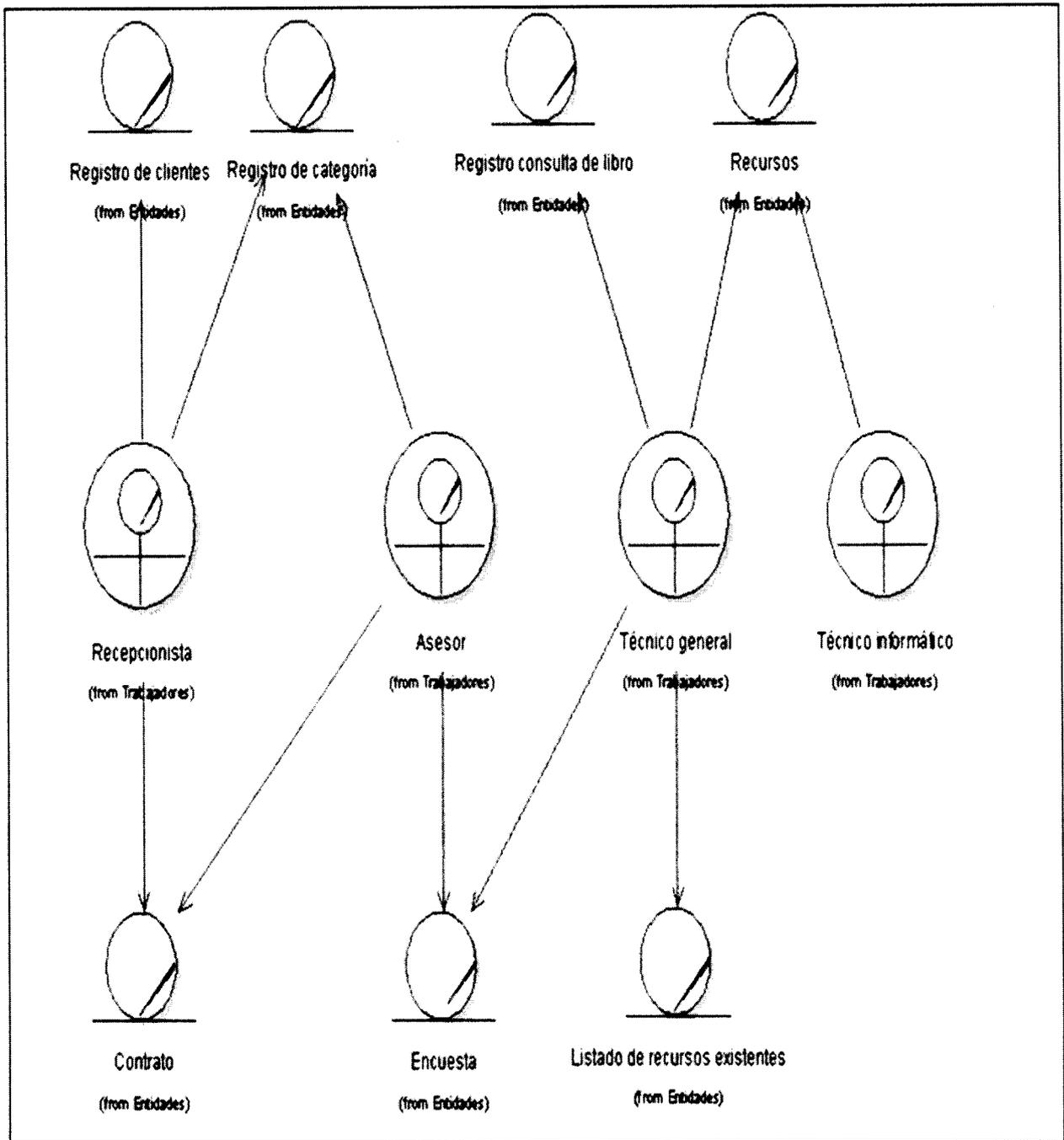


Figura 7: Diagrama de Modelo de Objetos del Negocio

3.3 Requerimientos Funcionales

Luego de ser analizado el problema de origen y teniendo en cuenta que los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir, fueron identificados los siguientes requisitos funcionales para el desarrollo del sistema propuesto:

R54: Administrar recurso

- R54.1: Insertar recurso.
- R54.2: Modificar recurso.
- R54.3: Eliminar recurso.

R55: Administrar clasificación

- R55.1: Crear clasificación.
- R55.2: Modificar clasificación.
- R55.3: Eliminar clasificación.

R70: Crear listado de recursos dado un tipo

3.4 Requerimientos No Funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Estas propiedades son características que hacen al producto atractivo, usable, rápido y confiable. Teniendo en cuenta el concepto anterior y las propiedades que debe tener el sistema que se investiga se tienen los siguientes requerimientos no funcionales:

Requerimientos de seguridad:

- **Confidencialidad:** La información manejada por el sistema deberá estar protegida de acceso no autorizado y divulgación. Los usuarios accederán a la información correspondiente a cada uno.
- **Integridad:** La información manejada por el sistema será objeto de cuidadosa protección contra estados inconsistentes y corrupción.
- **Firewall:** El servidor debe de implementar un firewall para la protección.

- Contraseña: Las contraseñas usadas en los servidores debe de ser de 10 o más caracteres y caducan en un período de 1 mes. Las contraseñas usadas por los usuarios de la aplicación debe ser de 7 o más caracteres y caducan en un período de un mes.

Requerimientos de Disponibilidad:

- A los usuarios autorizados se les deberá garantizar el acceso a la información solicitada en todo momento.
- Los recursos han de estar dispuestos virtualmente como mismo están dispuestos físicamente en los CASIE.
- Los recursos deben abrirse como libros virtuales.

Requerimientos de Interfaz:

- Como la aplicación propuesta será usada por personas que no necesariamente tienen habilidades en el trabajo con la computadora, debe ser una interfaz amigable, legible, interactiva, fácil de usar, profesional, clara, sencilla.
- Las respuestas de las tareas docentes deben indicarle al usuario si va progresando o no en la realización de las mismas.

Requerimientos de Usabilidad:

- Para utilizar la aplicación es necesario estar registrado en la aplicación según el rol disponible, para facilitar una mejor utilización de los datos y acceso de información.
- El módulo de administración debe permitir la petición concurrente de un gran número de solicitudes de los otros módulos.
- El tiempo y fecha del sistema en el servidor deben corresponderse con la fecha y hora reales.

Requerimientos de Software:

- Para la implantación del sistema se requiere de un servidor Windows NT.
- Para la utilización del sistema por lado del cliente se hace necesario tener disponible un navegador web compatible o superior con Internet Explorer.

- Se utiliza MySQL server V5.0.22 como servidor de bases de Datos.

Requerimientos de Hardware:

- Las computadoras que usarán los usuarios requieren como mínimo un procesador Pentium 2, 64 Mb de memoria RAM. Deben de estar conectadas en red con el servidor.
- Para los servidores (Web, SGBD) se hace necesario: PENTIUM II o superior con 512 MB de RAM o más. 4 2.0 GHz o superior, y 40Gb de capacidad de disco o más.

Requerimientos de Confiabilidad:

- Además de ser seguro es muy confiable ya que garantiza que los datos lleguen íntegros e intactos a su destino.
- Los datos son almacenados en una BD segura.

Requerimientos de Portabilidad:

- La aplicación podrá ser usada bajo el sistema operativo Windows.

Requerimientos de Rendimiento:

- El sistema debe ser lo más eficiente posible para poder lograr un tiempo de respuesta adecuado.

Requerimientos de soporte:

- Diccionarios digitales o enlaces a sitios de diccionarios.
- Garantizar la prueba del sistema.
- Dar un adiestramiento a las personas que tendrán la responsabilidad de administrar el sistema.

3.5 Descripción del Sistema Propuesto

3.5.1 Descripción de los actores del sistema

Actores del sistema	Justificación
Administrador del sistema	Es aquella persona que se encarga de mantener, registrar y controlar todos los eventos que ocurren en el sistema.

Tabla 10: Justificación de los actores del sistema

3.5.2 Casos de Uso del Sistema

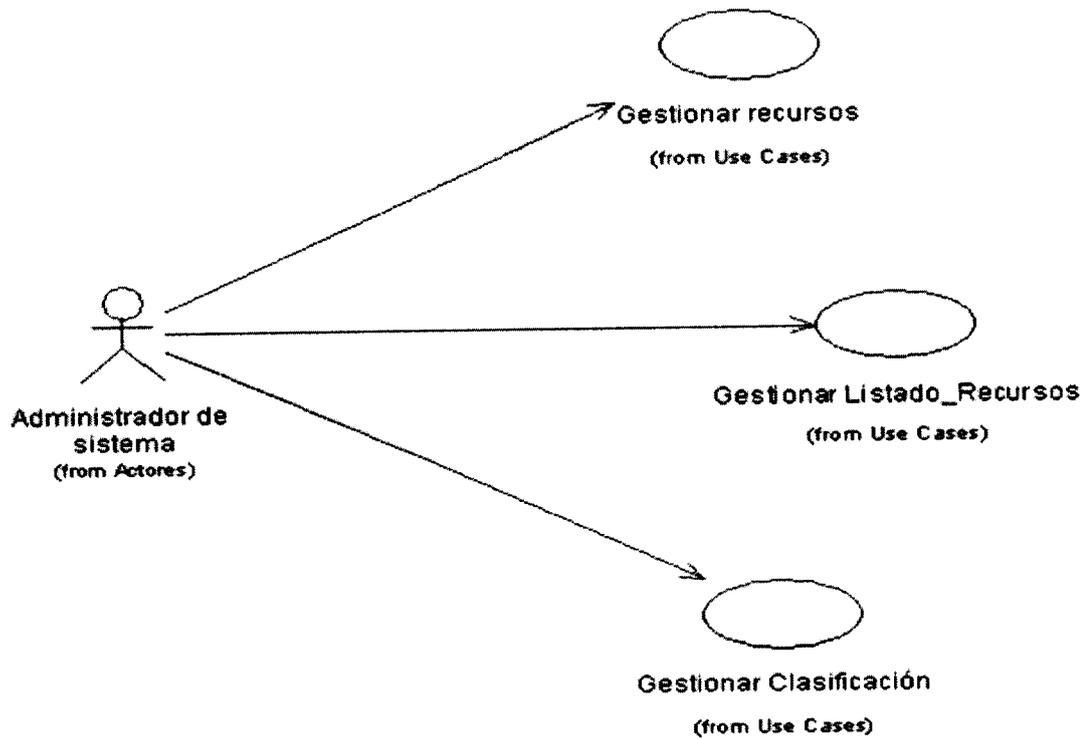


Figura 8: Diagrama de Casos de Uso del Sistema

Caso de Uso:	Gestionar recurso.	
Actores:	Administrador (inicia).	
Propósito	Permite al administrador del sistema gestionar toda la información referente a un recurso (Ordenar los recursos utilizados por el usuario en orden descendiente, Listar los recursos didácticos pertenecientes al servicio recursos).	
Resumen:	El CUS se inicia cuando el administrador del sistema selecciona la opción de Gestionar recurso, luego escoge el tipo de gestión que desea, introduce los datos necesarios y el sistema realiza la acción seleccionada por el administrador del sistema terminando el CUS.	
Referencia	R 5, R 23, R 54, R 54.1, R 54.2, R 54.3.	
Precondiciones:	El administrador del sistema debe de estar autenticado.	
Poscondiciones	Recurso adicionado a la Base de Datos. Recurso eliminado de la Base de Datos. Recurso modificado en la Base de Datos.	
Flujo Normal de Eventos		
Sección ""		
Acción del Actor	Respuesta del Sistema	
1- El administrador del sistema selecciona la opción de "Gestionar recurso".	2- El sistema muestra las opciones: Ordenar los recursos utilizados por el usuario en orden descendiente, Listar los recursos didácticos pertenecientes al servicio recursos, Administrar recurso, Insertar recurso, Modificar recurso, Eliminar recurso, Procesar solicitud de recurso y generar respuesta, Mostrar recurso solicitado.	
Escenario 1: Ordenar los recursos utilizados por el usuario en orden descendiente		
1- El administrador del sistema selecciona la opción de ordenar los recursos utilizados por el usuario en orden descendiente.	2- El sistema muestra un formulario para que el administrador del sistema escriba el nombre del usuario al cual quiere listar los recursos que utilizo.	
3- El administrador del sistema introduce el nombre del usuario.	4- El sistema verifica que el usuario introducido por el cliente exista en la Base de Datos.	
	5- Si el usuario existe, el sistema busca en la Base de Datos todos los recursos utilizados por este usuario, sino ir a flujo alterno 1.	
	6- El sistema lista de forma descendiente en cuanto a la fecha de utilización todos los recursos utilizados por el usuario.	
	7- El sistema le muestra al administrador del sistema la lista	

Capítulo 3: Presentación de la solución propuesta

	de recursos ya ordenada y termina el CUS.
Flujos Alternos 1	
Acción del Actor	Respuesta del Sistema
	1- Si el usuario no existe, el sistema le muestra al administrador del sistema un mensaje de que no existe y le indica retornar a la acción 3.
Escenario 2: Listar los recursos didácticos pertenecientes al servicio recursos.	
1- El administrador del sistema selecciona la opción de Listar los recursos didácticos pertenecientes al servicio recursos.	2- El sistema busca en la Base de Datos todos los recursos que existan.
	3- El sistema le muestra al administrador del sistema la lista de todos los recursos didácticos y termina el CUS.
Escenario 3: Administrar recurso.	
Escenario 4: Insertar recurso.	
1- El administrador del sistema selecciona la opción de Insertar recurso.	2- El sistema muestra un formulario a completar para insertar un nuevo recurso.
3- El administrador del sistema introduce los datos solicitados por el sistema y presiona el botón "Insertar".	4- El sistema verifica los datos introducidos por el administrador del sistema.
	5- Si los datos introducidos son correctos, el sistema comprueba si este recurso ya existe en la Base de Datos, sino ir a flujo alternativo 1.
	6- Si el recurso no está en la Base de Datos, el sistema lo agrega en la Base de Datos y termina el CUS, sino ir a flujo alternativo 2.
Flujos Alternos 1	
Acción del Actor	Respuesta del Sistema
	1- Si los datos introducidos por el administrador del sistema son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato incorrecto e indica al administrador del sistema retornar a la acción 3.
Flujos Alternos 2	
Acción del Actor	Respuesta del Sistema
	1- Si el recurso ya existe en la Base de Datos, el sistema le muestra al administrador del sistema un mensaje de que ya

Capítulo 3: Presentación de la solución propuesta

	existe y le alerta que si quiere sobrescribir el recurso.
2- El administrador del sistema elige si sobrescribir el recurso o no.	3- Si administrador del sistema eligió sobrescribir el recurso, el sistema sobrescribe el recurso y termina el CUS., sino ir a flujo alterno 3.
Flujos Alternos 3	
Acción del Actor	Respuesta del Sistema
	1- Si administrador del sistema eligió no sobrescribir el recurso, el sistema no ejecuta ninguna acción y termina el CUS.
Escenario 5: Modificar recurso.	
1- El administrador del sistema selecciona la opción de Modificar recurso.	2- El sistema muestra un listado con los recursos existentes en la Base de Datos.
3- El administrador del sistema selecciona el recurso que quiere modificar.	4- El sistema localiza los datos del recurso y los muestra listo para ser modificados.
5- El administrador del sistema realiza los cambios necesarios en los datos y presiona el botón "Modificar".	6- El sistema verifica que los datos son entrados correctamente.
	7- Si los datos introducidos son correctos, el sistema actualiza los datos del recurso en la Base de Datos y termina el CUS, sino ir a flujo alterno 1.
Flujos Alternos 1	
Acción del Actor	Respuesta del Sistema
	1- Si los datos introducidos por el administrador del sistema son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato incorrecto e indica al administrador del sistema retornar a la acción 5.
Escenario 6: Eliminar recurso.	
1- El administrador del sistema selecciona la opción de Eliminar recurso.	2- El sistema muestra una lista con todos los recursos existentes en la Base de Datos.
3- El administrador del sistema selecciona el recurso que quiere eliminar y da en el botón "Eliminar".	4- El sistema muestra un mensaje de advertencia para dicha acción.
5- El administrador del sistema confirma si desea eliminar o no el recurso.	6- Si el administrador del sistema acepta, el sistema elimina el recurso seleccionado de la Base de Datos y termina el CUS, sino ir a flujo alterno 1.

Flujos Alternos 1	
Acción del Actor	Respuesta del Sistema
	1- Si el administrador del sistema cancela la acción, el sistema no ejecuta ninguna acción y termina el CUS.
Escenario 7: Procesar solicitud de recurso y generar respuesta.	
1- El usuario selecciona la opción de Procesar solicitud de recurso y generar respuesta.	2- El sistema busca en la base de datos los recursos existentes.
	3- Muestra un listado de los recursos existentes en el sistema.
Escenario 8: Mostrar recurso solicitado.	
1- El administrador del sistema selecciona la opción de Mostrar recurso solicitado.	2- El sistema verifica que exista el recurso y que este haya sido solicitado.
	3- Si el recurso fue solicitado, muestra en una interfaz el recurso y termina el CUS, sino ir a flujo alternativo 1.
Flujos Alternos 1	
Acción del Actor	Respuesta del Sistema
	1- Si el recurso no fue solicitado, muestra un mensaje de error y no ejecuta la acción.
Flujos Alternos 1	
Acción del Actor	Respuesta del Sistema
1- Si el usuario no encuentra el recurso que quiere ver, presiona el botón "Cancelar".	2- El sistema cierra la ventana con la lista de recurso y termina el CUS.
Prioridad	Crítico.
Prototipo de Interfaz	

Tabla 11: Descripción Textual del Casos de Uso Gestionar Recurso

Caso de Uso:	Gestionar clasificación.
Actores:	Administrador (inicia).
Propósito	Permite al administrador del sistema gestionar toda la información referente a la clasificación de un recurso (Administrar clasificación, Crear clasificación, Modificar clasificación, Eliminar clasificación).
Resumen:	El CUS se inicia cuando el administrador del sistema selecciona la opción de Gestionar clasificación, luego escoge el tipo de gestión que desea, introduce los datos necesarios y el sistema realiza la acción seleccionada por el administrador del sistema terminando el CUS.
Referencia	R 55, R 55.1, R 55.2, R 55.3.

Capítulo 3: Presentación de la solución propuesta

Precondiciones:	El administrador del sistema debe de estar autenticado.	
Poscondiciones	<ol style="list-style-type: none"> 1. Clasificación creada en la Base de Datos. 2. Clasificación eliminada en la Base de Datos. 3. Clasificación modificada en la Base de Datos. 	
Flujo Normal de Eventos		
Sección ""		
Acción del Actor	Respuesta del Sistema	
1- El administrador del sistema selecciona la opción de "Gestionar clasificación".	2- El sistema muestra las opciones: Administrar clasificación, Crear clasificación, Modificar clasificación, Eliminar clasificación.	
Escenario 1: Administrar clasificación.		
1- El administrador del sistema selecciona la opción de Administrar clasificación.		
Escenario 2: Crear clasificación.		
1- El administrador del sistema selecciona la opción de Crear clasificación.	2- El sistema muestra un formulario a completar para insertar una nueva clasificación.	
3- El administrador del sistema introduce los datos solicitados por el sistema y presiona el botón "crear".	4- El sistema verifica los datos introducidos por el administrador del sistema.	
	5- Si los datos introducidos son correctos, el sistema comprueba si esta clasificación ya existe en la Base de Datos, sino ir a flujo alterno 1.	
	6- Si la clasificación no está en la Base de Dato, el sistema lo agrega en la Base de Datos y termina el CUS, sino ir a flujo alterno 2.	
Flujos Alternos 1		
Acción del Actor	Respuesta del Sistema	
	1- Si los datos introducidos por el administrador del sistema son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato incorrecto e indica al administrador del sistema retornar a la acción 3.	
Flujos Alternos 2		
Acción del Actor	Respuesta del Sistema	
	1- Si la clasificación ya existe en la Base de Datos, el sistema le muestra al administrador del sistema un mensaje de que ya	

Capítulo 3: Presentación de la solución propuesta

	existe y le indica retornar a la acción 3.
Escenario 3: Modificar clasificación.	
1- El administrador del sistema selecciona la opción de Modificar clasificación.	2- El sistema muestra un listado con las clasificaciones existentes en la Base de Datos.
3- El administrador del sistema selecciona la clasificación que quiere modificar.	4- El sistema localiza los datos de la clasificación y los muestra listo para ser modificados.
5- El administrador del sistema realiza los cambios necesarios en los datos y presiona el botón "Modificar".	6- El sistema verifica que los datos son entrados correctamente.
	7- Si los datos introducidos son correctos, el sistema actualiza los datos de la clasificación en la Base de Datos y termina el CUS, sino ir a flujo alternativo 1.
Flujos Alternos 1	
Acción del Actor	Respuesta del Sistema
	1- Si los datos introducidos por el administrador del sistema son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato incorrecto e indica al administrador del sistema retornar a la acción 5.
Escenario 4: Eliminar clasificación.	
1- El administrador del sistema selecciona la opción de Eliminar clasificación.	2- El sistema muestra una lista con todas las clasificaciones existentes en la Base de Datos.
3- El administrador del sistema selecciona la clasificación que quiere eliminar y da en el botón "Eliminar".	4- El sistema muestra un mensaje de advertencia para dicha acción.
5- El administrador del sistema confirma si desea eliminar o no la clasificación.	6- Si el administrador del sistema acepta, el sistema elimina la clasificación seleccionada de la Base de Datos y termina el CUS, sino ir al flujo alternativo 1.
Flujos Alternos 1	
Acción del Actor	Respuesta del Sistema
	1- Si el administrador del sistema cancela la acción, el sistema no ejecuta ninguna acción y termina el CUS.
Prioridad	Crítico.
Prototipo de Interfaz	

Tabla 12: Descripción Textual del Casos de Uso Gestionar clasificación

Caso de Uso:	Gestionar listado de recursos.	
Actores:	Administrador (inicia).	
Propósito	Permite al administrador del sistema gestionar toda la información referente a un recurso (Crear listado de recursos dado un tipo, Permitir al usuario seleccionar un recurso).	
Resumen:	El CUS se inicia cuando el administrador del sistema selecciona la opción de Gestionar listado de recursos, luego escoge el tipo de gestión que desea, introduce los datos necesarios y el sistema realiza la acción seleccionada por el administrador del sistema terminando el CUS.	
Referencia	R 70 y R 71.	
Precondiciones:	El administrador del sistema debe de estar autenticado.	
Poscondiciones		
Flujo Normal de Eventos		
Sección ""		
Acción del Actor	Respuesta del Sistema	
1- El administrador del sistema selecciona la opción de "Gestionar listado de recursos".	2- El sistema muestra las opciones: Crear listado de recursos dado un tipo, Permitir al usuario seleccionar un recurso didáctico.	
Escenario 1: Crear listado de recursos dado un tipo.		
1- El administrador del sistema selecciona la opción de Crear listado de recursos dado un tipo.	2- El sistema muestra un formulario para que el administrador del sistema escriba el tipo de recurso del cual quiere listar los recursos existentes.	
3- El administrador del sistema introduce el tipo de recurso.	4- El sistema verifica que ese tipo de recurso introducido por el cliente exista en la Base de Datos.	
	5- Si el tipo de recurso existe, el sistema busca en la Base de Datos todos los recursos que pertenezcan a ese tipo, sino ir a flujo alternativo 1.	
	6- El sistema le muestra al administrador del sistema la lista de recursos que pertenecen a ese tipo y termina el CUS.	
Flujos Alternos 1		
Acción del Actor	Respuesta del Sistema	
	1- Si el tipo de recurso no existe, el sistema le muestra al administrador del sistema un mensaje de que no existe y le indica retornar a la acción 3.	
Escenario 2: Permitir al usuario seleccionar un recurso didáctico.		

1- El usuario selecciona la opción de seleccionar un recurso didáctico.	2- El sistema le muestra un listado con todos los recursos didácticos existentes en la Base de Datos.
3- El usuario busca en la lista, el recurso que quiere ver.	
4- Si el usuario encuentra el recurso que desea ver, lo selecciona y presiona el botón "Ver", sino ir a flujo alterno 1.	5- El sistema le muestra el recurso didáctico al usuario y termina el CUS.
Flujos Alternos 1	
Acción del Actor	Respuesta del Sistema
1- Si el usuario no encuentra el recurso que quiere ver, presiona el botón "Cancelar".	2- El sistema cierra la ventana con la lista de recurso y termina el CUS.
Prioridad	Crítico.
Prototipo de Interfaz	

Tabla 13: Descripción Textual del Casos de Uso Gestionar listado de recursos

3.6 Conclusiones parciales

En este capítulo se desarrolló el modelamiento del negocio para una mejor comprensión del problema a resolver. Se analizaron los requerimientos funcionales y no funcionales del sistema, respetando las solicitudes y necesidades del cliente. Además se realizó la descripción textual de todos los casos de usos.

Capítulo 4: Construcción de la solución propuesta

4.1 Introducción

En este capítulo se realiza el análisis y consecutivamente el diseño del sistema modelando los diagramas de clases de análisis y diseño, donde se desarrollan los Diagrama de Clases de Diseño, el Modelo de Clases Persistente, el Modelo de Datos y el Modelo de Despliegue. También se analizan los principios del diseño, donde se hace referencia a los estándares de interfaz.

4.2 Arquitectura

La arquitectura de software se define como un conjunto de patrones abstractos (ejemplos) que permiten dirigir y coordinar grandes sistemas de software. A través de una arquitectura se pueden desarrollar los conceptos y los planes de descomposición del software en unidades denominadas módulos. También se puede definir en forma ordenada la interacción y dependencia de cada módulo involucrado en la resolución del problema, además se definen las interfaces de uso y de comunicación con sistemas externos. Con una arquitectura clara, se pueden definir características de diseño innovadoras, operaciones sobre el negocio, lógica y flujos de alto nivel.(4)

La arquitectura de una aplicación es la vista conceptual de la estructura de ésta. Toda aplicación contiene código de presentación, código de procesamiento de datos y código de almacenamiento de datos.(28)

Tipos de arquitecturas más conocidas:

- **Arquitectura centralizada:** Se basa en la existencia de una máquina servidora que almacena los datos y las aplicaciones que los procesan. Los clientes se comportan como terminales y solo sirven para introducir datos desde teclado.

Ventajas:	Desventajas:
Fácil de administrar.	Alto coste.
Gran nivel de seguridad.	Máquina servidora muy cargada.

- **Arquitectura cliente/servidor:** Se basa en la existencia de dos tipos de aplicaciones ejecutándose de forma independiente. Una de las aplicaciones actúa como cliente y otra como servidora.

Ventajas:	Desventajas:
Fácil de escalar.	Nuevas aplicaciones.
Reparto de cargas.	Importancia de las comunicaciones.

(29)

4.2.1 Arquitectura en tres capas

La programación por capas es un estilo de programación en el que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario.

La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado.

En la arquitectura en 3 capas, intervienen la capa de Presentación, la de Lógica y de Acceso a datos, donde:

- **Presentación:** Son los formularios e inclusive código fuente que valide la entrada de datos en esos formularios, como ser alguna función que no permita introducir solo números.
- **Lógica:** Esta la parte más compleja e incluye todo lo que el programa debe hacer, aquí también se arman las consultas SQL para acceder a datos que es enviada a la siguiente capa (Acceso a datos).
- **Acceso a datos:** Esta parte es la encargada de acceder a los datos de una base de datos, realizando cualquier operación contra esta misma como ser Insertar, Modificar Eliminar.

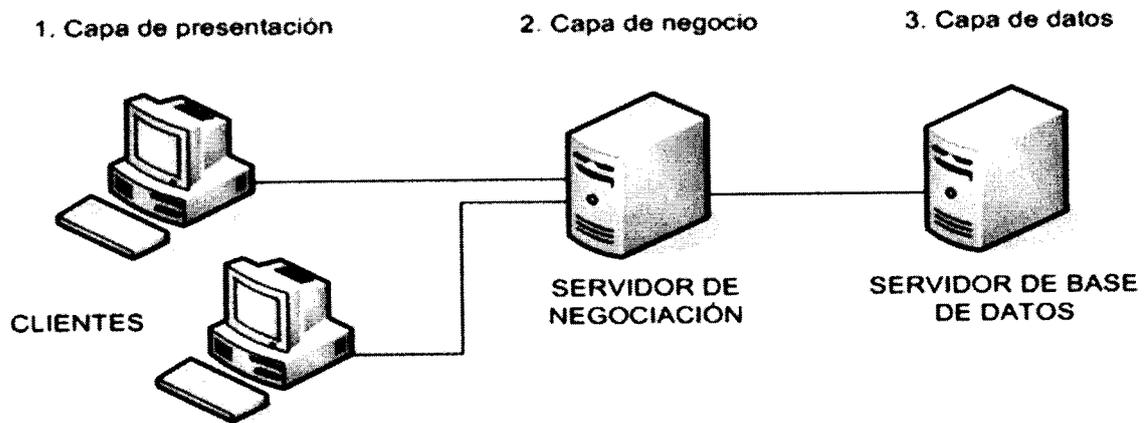


Figura 9: Arquitectura en 3capas

Como se podría esperar cada uno de los componentes de la aplicación en una arquitectura 3 capas se separa en una sola entidad. Esto te permite implementar componentes de una manera más flexible. Algo que no creo que sorprenda es la afirmación de que este tipo de arquitectura es la más compleja.

En esta arquitectura todas las peticiones de los clientes se controlan en la capa correspondiente a la lógica del negocio. Cuando el cliente necesita hacer una petición se la hace a la capa en la que se encuentra la lógica del negocio. Esto es bastante importante pues eso quiere decir que:

- El cliente no tiene que tener drivers ODBC ni la problemática consiguiente de instalación de los drivers por tanto se reduce el costo de mantener las aplicaciones cliente
- El Cliente y el Gestor de Reglas de negocio tienen que hablar el mismo lenguaje (en este caso COM)
- El Gestor de Reglas de Negocio y el Servidor de Datos tienen que hablar el mismo lenguaje (en este caso ODBC)
- Lo ideal sería que el Gestor de Reglas de Negocio no sólo OLE y ODBC sino otros estándares como DBLib, OLI, DRDA, SQL/API y X/Open.

La arquitectura de tres capas ha logrado gran alcance a nivel mundial ya que es una arquitectura que funciona bastante bien en problemas de mediana complejidad y permite ser el referente inicial en la concepción o primeras etapas (fase de inicio) del desarrollo de un sistema de software.(30)

4.2.2 Patrones

En la tecnología de objetos un patrón es una descripción del problema y su solución, a la que se le da un nombre y que se puede aplicar a nuevos contextos.

Los patrones GRASP, (patrones generales de software para asignar responsabilidades) describen los principios de diseño de objetos para la asignación de responsabilidades.

En cuanto a las responsabilidades UML define una responsabilidad como un contrato u obligación de un clasificador:

- Las responsabilidades están relacionadas con las obligaciones de un objeto en cuanto a su comportamiento.
- Las responsabilidades se relacionan con las obligaciones de un objeto respecto de su comportamiento. Estas responsabilidades pertenecen, esencialmente, a dos categorías: conocer y hacer.

Entre las responsabilidades de un objeto relacionadas con el hacer se encuentran:

- Hacer algo en uno mismo.
- Iniciar una acción en otros objetos.
- Controlar y coordinar actividades en otros objetos.
- Entre las responsabilidades de un objeto relacionadas con el conocer se encuentran:
- Estar enterado de los datos privados encapsulados.
- Estar enterado de la existencia de objetos conexos.
- Estar enterado de cosas que se pueden derivar o calcular.

Se destacan 5 patrones fundamentales:

1. Experto
2. Creador
3. Bajo Acoplamiento
4. Alta Cohesión
5. Controlador(31)

1. Experto: La responsabilidad de realizar una labor es de la clase que tiene o puede tener datos involucrados (atributos). Una clase contiene toda la información necesaria para realizar la labor que tiene encomendada. Hay que tener en cuenta que esto es aplicable mientras se consideren los mismos aspectos del sistema:

1. Lógica de negocio
2. Persistencia a la base de datos
3. Interfaz de usuario

2. Creador: Se asigna la responsabilidad de que una clase B cree un objeto de la clase A solamente cuando:

1. B contiene a A.
2. B es una agregación (o composición) de A.
3. B almacena a A.
4. B tiene los datos de inicialización de A (datos que requiere su constructor).
5. B usa a A.

La creación de instancias es una de las actividades más comunes en un sistema orientado a objetos. En consecuencia es útil contar con un principio general para la asignación de las responsabilidades de creación. Si se asignan bien el diseño puede soportar un bajo acoplamiento, mayor claridad, encapsulación y reutilización.

3. Bajo acoplamiento: Debe haber pocas dependencias entre las clases. Si todas las clases dependen de todas ¿Cuánto software se puede extraer de un modo independiente y reutilizarlo en otro proyecto? Para determinar el nivel de acoplamiento de clases, son muy buenos los diagramas de colaboración de UML. Uno de los principales síntomas de un mal diseño y alto acoplamiento es una herencia muy profunda. Siempre hay que considerar las ventajas de la delegación respecto de la herencia.

4. Alta Cohesión: Cada elemento del diseño debe realizar una labor única dentro del sistema, no desempeñada por el resto de los elementos y auto-identificable.

Ejemplos de una baja cohesión son clases que hacen demasiadas cosas. En todas las metodologías la refactorización. Uno de los elementos a refactorizar son las clases saturadas de métodos. Un buen ejemplo de diseño se produce cuando se crean los denominados paquetes de servicio o clases

agrupadas por funcionalidades que son fácilmente reutilizables (bien por su uso directo o por herencia).

5. Controlador: Asignar la responsabilidad de controlar el flujo de eventos del sistema, a clases específicas. Esto facilita la centralización de actividades (validaciones, seguridad, etc.). El controlador no realiza estas actividades, las delega con otras clases con las que mantiene un modelo de alta cohesión. Un error muy común es asignarle demasiada responsabilidad y alto nivel de acoplamiento con el resto de los componentes del sistema.(32)

4.3Análisis

El análisis del sistema consiste en obtener una visión del sistema que se quiere obtener, de modo que sólo se interesa por los requisitos funcionales, transformar los requisitos funcionales en un diseño de clases viendo las relaciones e interacción que existe entre ellos, y teniendo en cuenta en el proceso una arquitectura robusta que permita adaptar el sistema al entorno de implementación que se está desarrollando.

4.3.1 Diagramas de Clases del análisis

Un diagrama de clases muestra un conjunto de clases, interfaces, y colaboraciones y sus relaciones. Gráficamente un diagrama de clase es una colección de vértices y arcos. Un diagrama de clase es justo un tipo especial de diagrama y comparte propiedades comunes al igual que todos los otros diagramas, un nombre y un contenido gráfico son una proyección dentro de un modelo. Son los diagramas más comunes en el modelado de sistemas orientados a objetos. Los diagramas de clase son importantes no solo para la visualización, especificación y documentación del modelo estructural, también para la construcción de sistemas ejecutables. Ingeniería hacia adelante e ingeniería inversa.(33)

Un diagrama de clases del análisis es un artefacto en el que se representan los conceptos en un dominio del problema. Representa las cosas del mundo real, no de la implementación automatizada de estas cosas.

En una aplicación donde la arquitectura es en tres capas como la que se está desarrollando, en la capa de presentación se encuentran las clases interfaz porque es donde se ejecutan las aplicaciones del cliente, en la capa de lógica del negocio las clases de control porque es donde ocurren todos los procesos y en la capa de acceso a datos se encuentran las clases entidad ya que allí es donde se accede a la base de datos.

A continuación se representan los diagramas de clases del análisis del problema que se investiga:

4.3.1.1 Diagrama de Clases del Análisis CU Gestionar clasificación

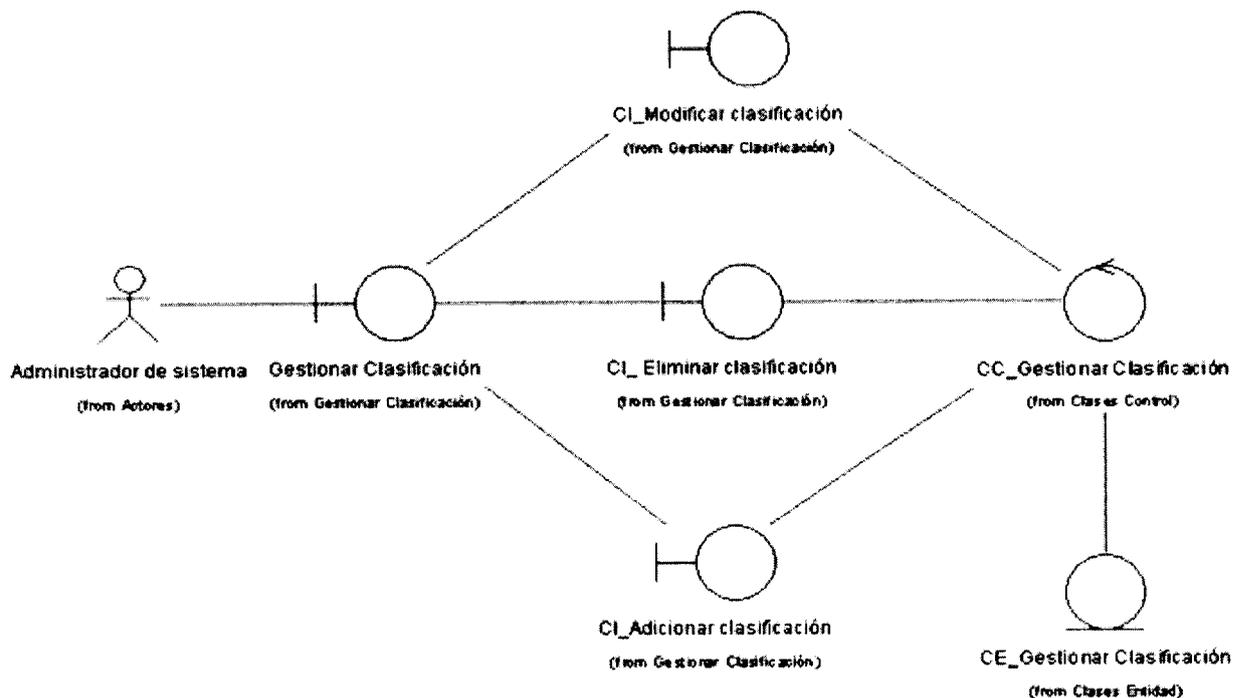


Figura 10: Diagrama de Clases del Análisis CU Gestionar clasificación

4.3.1.2 Diagrama de Clases del Análisis CU Gestionar listado de recursos

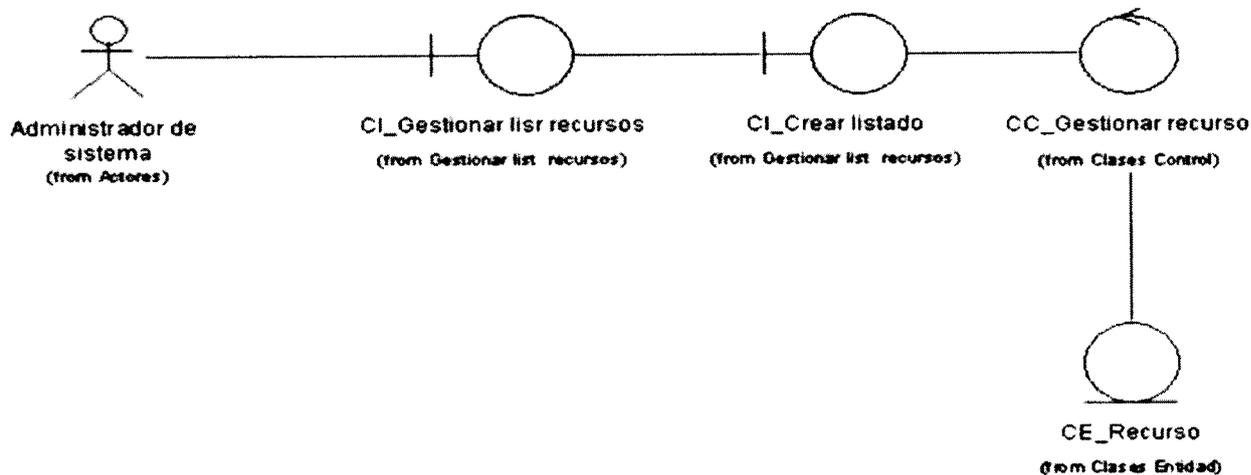


Figura 11: Diagrama de Clases del Análisis CU Gestionar listado de recursos

4.3.1.3 Diagrama de Clases del Análisis CU Gestionar recursos

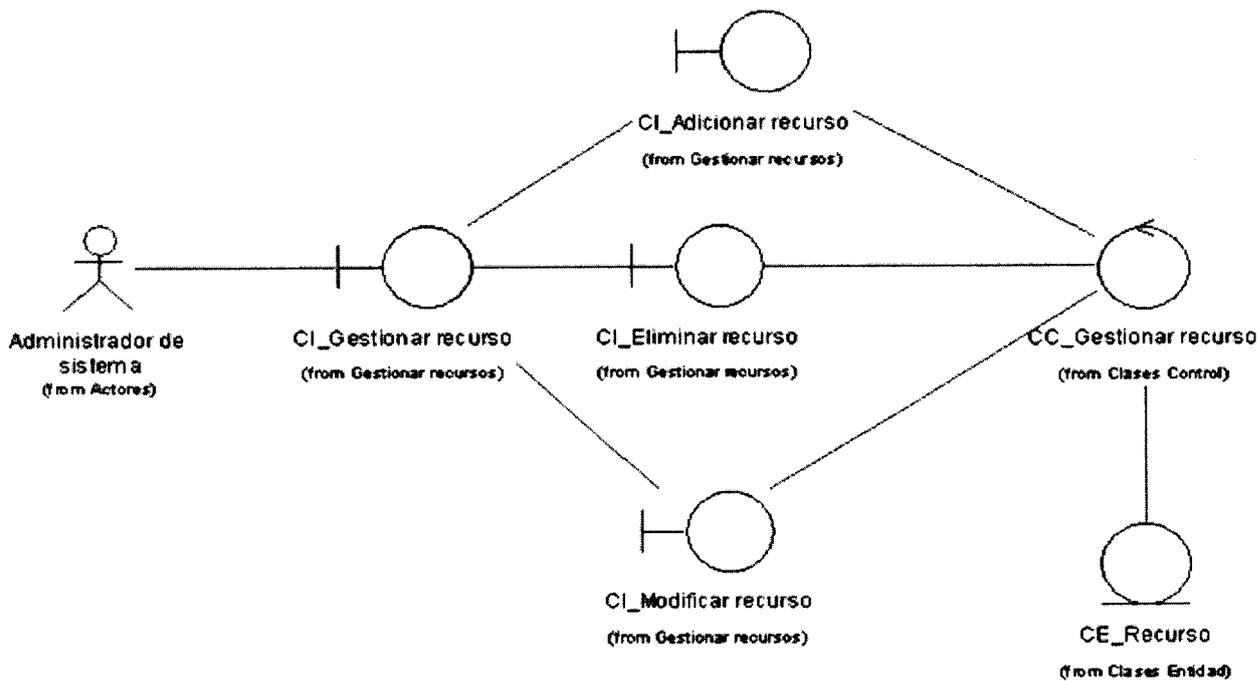


Figura 12: Diagrama de Clases del Análisis CU Gestionar recursos

4.4 Diseño

El diseño permite la comprensión de los aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, Sistemas Operativos, tecnologías de distribución y concurrencia y tecnologías de interfaz de usuario y se crea una entrada apropiada y un punto de partida para actividades de implementación, capturando los requisitos o subsistemas individuales, interfaces y clases. Una de las principales salidas del diseño lo constituyen el Diagrama de Clases del Diseño, el Modelo de Datos y el Diagrama de Despliegue.

Los diagramas de clases Web fueron divididos por casos de uso para una mejor comprensión a la hora de ser programada la aplicación, siendo una guía estructurada por empaquetamiento de Casos de Uso del sistema.

4.4.1 Diagrama de diseño del CU Gestionar listado de recurso

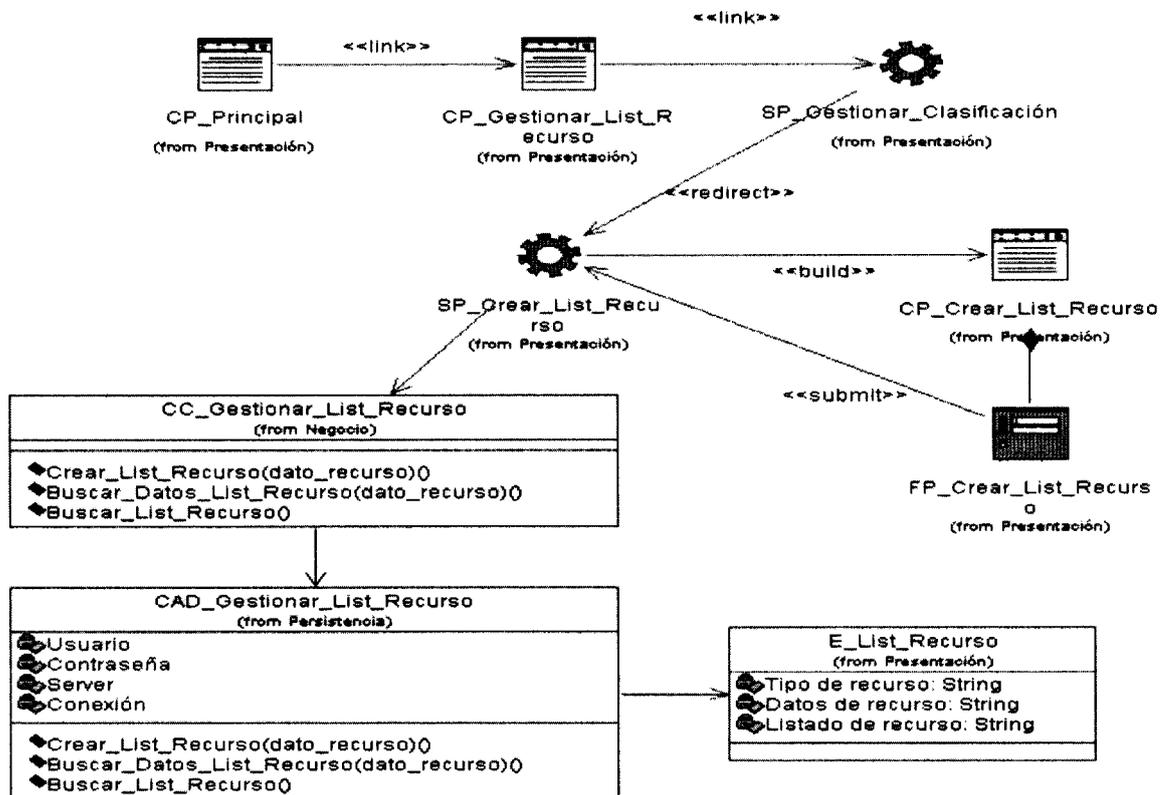


Figura 13: Diagrama de diseño del CU Gestionar listado de recurso

4.4.2 Diagrama de diseño del CU Gestionar recurso

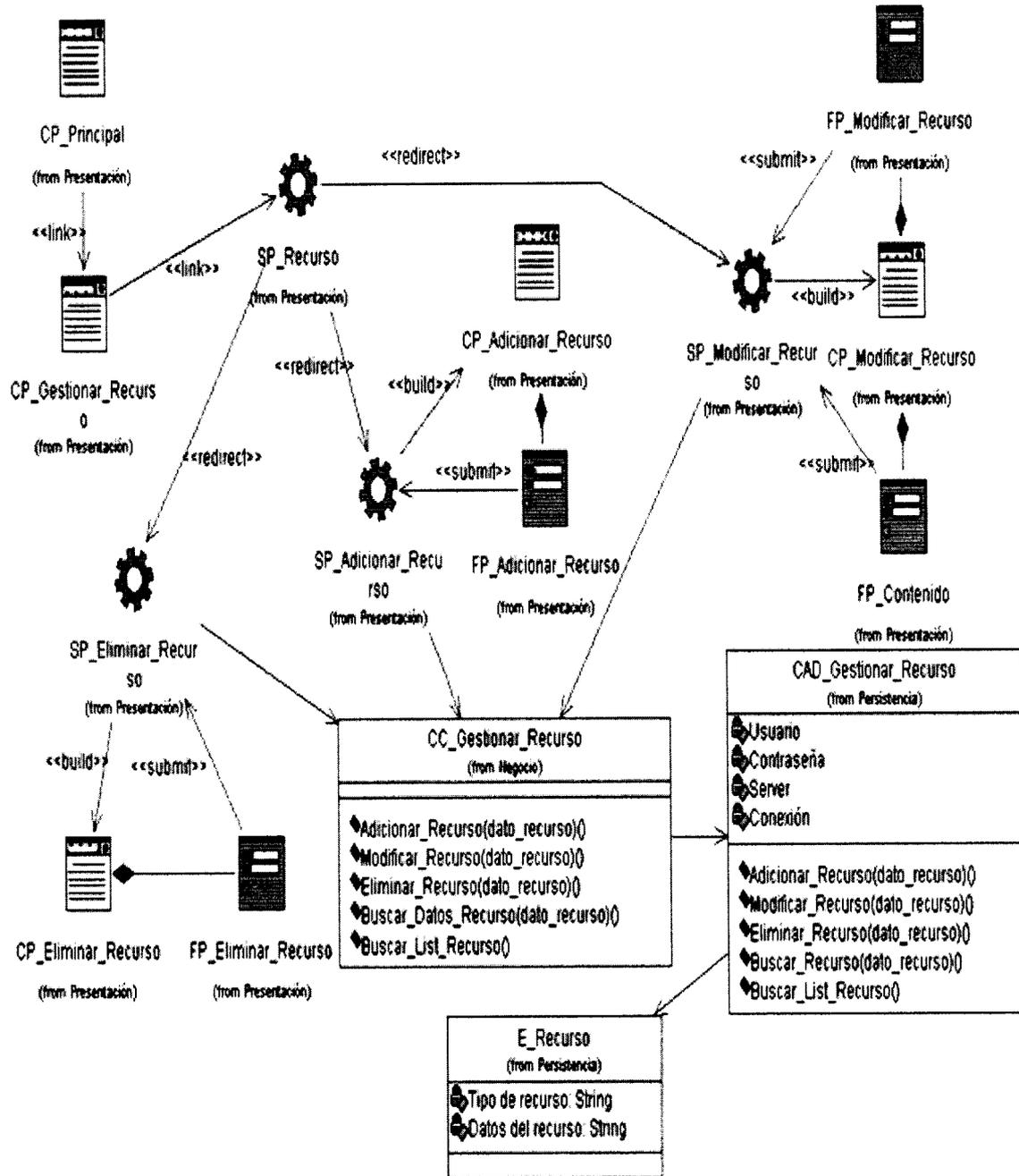


Figura 14: Diagrama de diseño del CU Gestionar recurso

4.4.3 Diagrama de diseño del CU Gestionar clasificación

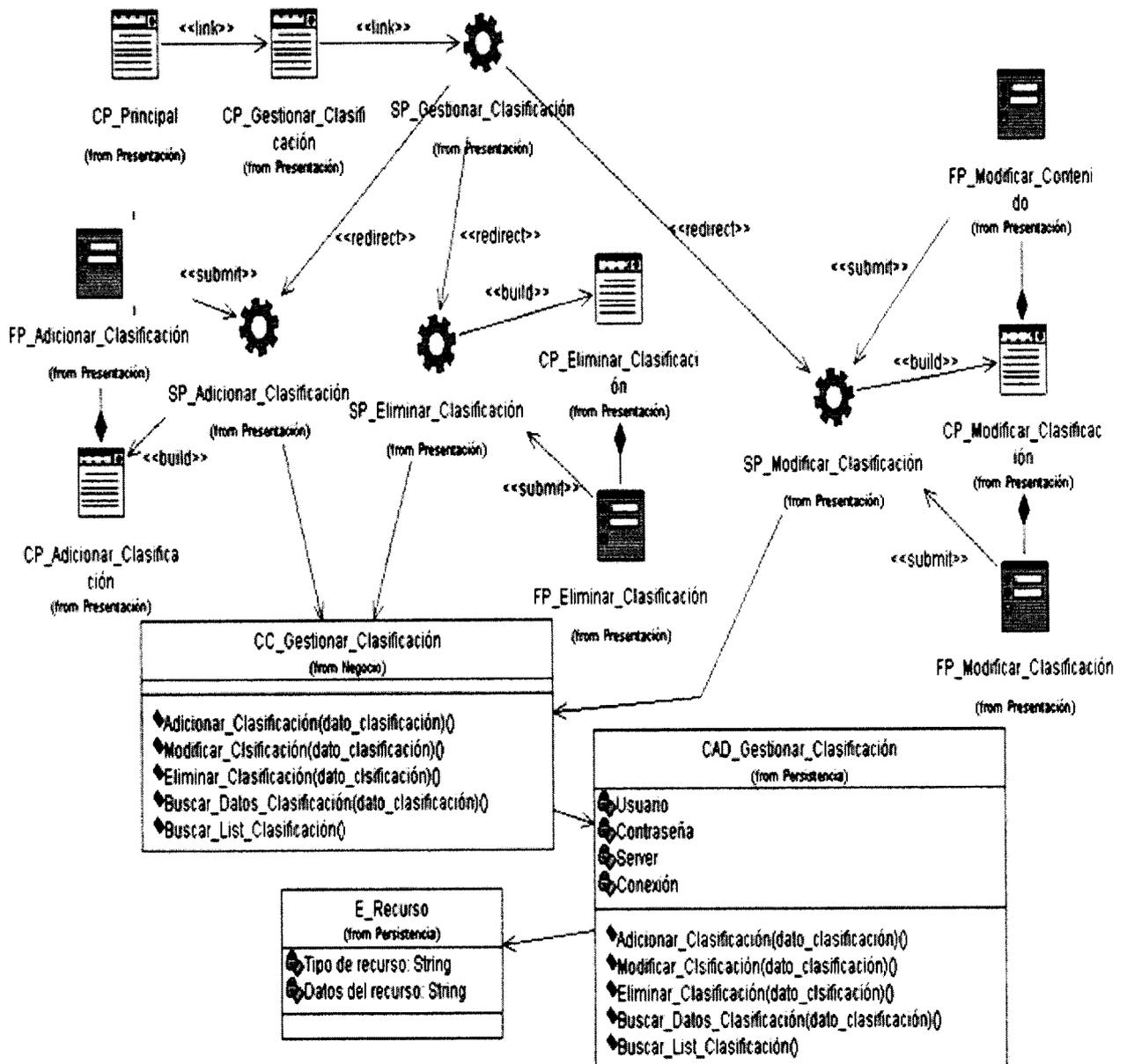


Figura 15: Diagrama de diseño del CU Gestionar clasificación

4.5 Modelo de Despliegue

Un diagrama de despliegue muestra las relaciones físicas entre los componentes de hardware y de software en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes software (procesos y objetos que se ejecutan en ellos).

Estarán conformados por instancias de los componentes software que se representan manifestaciones del código en tiempo de ejecución (los componentes que sólo sean utilizados en tiempo de compilación deben mostrarse en el diagrama de componentes). Un diagrama de despliegue es un grafo de nodos unidos por conexiones de comunicación. Un nodo puede contener instancias de componentes de software, objetos, procesos (caso particular de un objeto). En general un nodo será una unidad de computación de algún tipo, desde un sensor a un mainframe. Las instancias de componentes software pueden estar unidas por relaciones de dependencia, posiblemente a interfaces (ya que un componente puede tener más de una interfaz).

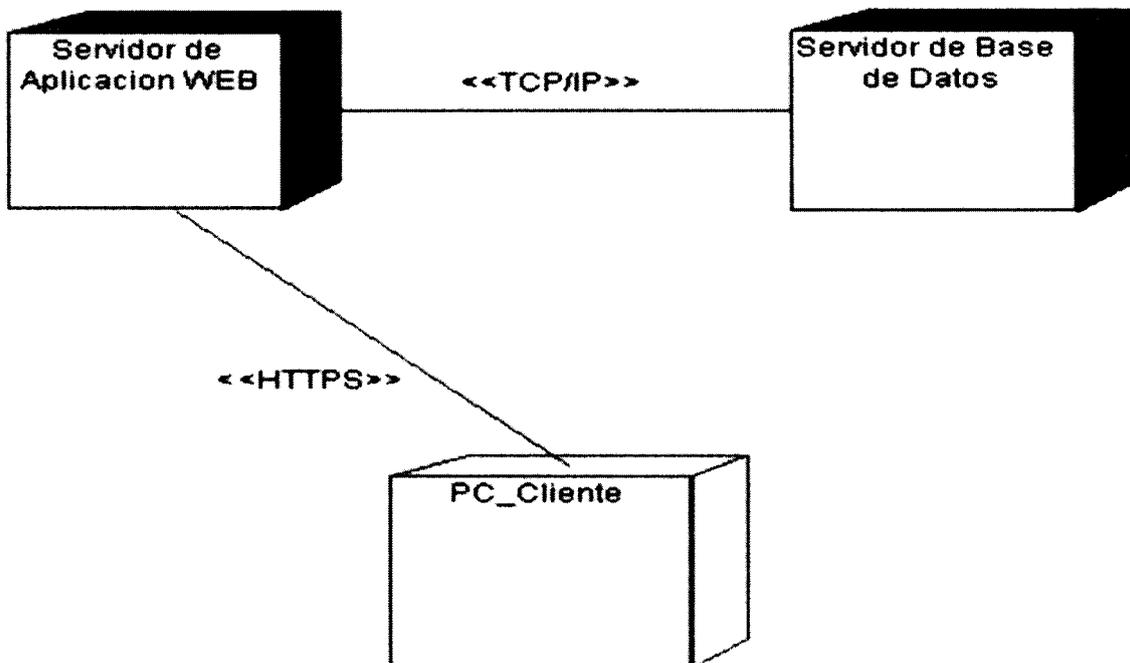


Figura 16: Diagrama de Modelo de Despliegue

4.6 Modelo de Implementación

Un diagrama de componentes muestra las dependencias lógicas entre componentes software, sean éstos, componentes fuentes, binarios o ejecutables. Los componentes software tienen tipo, que indica si son útiles en tiempo de compilación, enlace o ejecución.

Seguidamente se representan los diagramas de componentes del sistema que se investiga divididos por Casos de Uso.

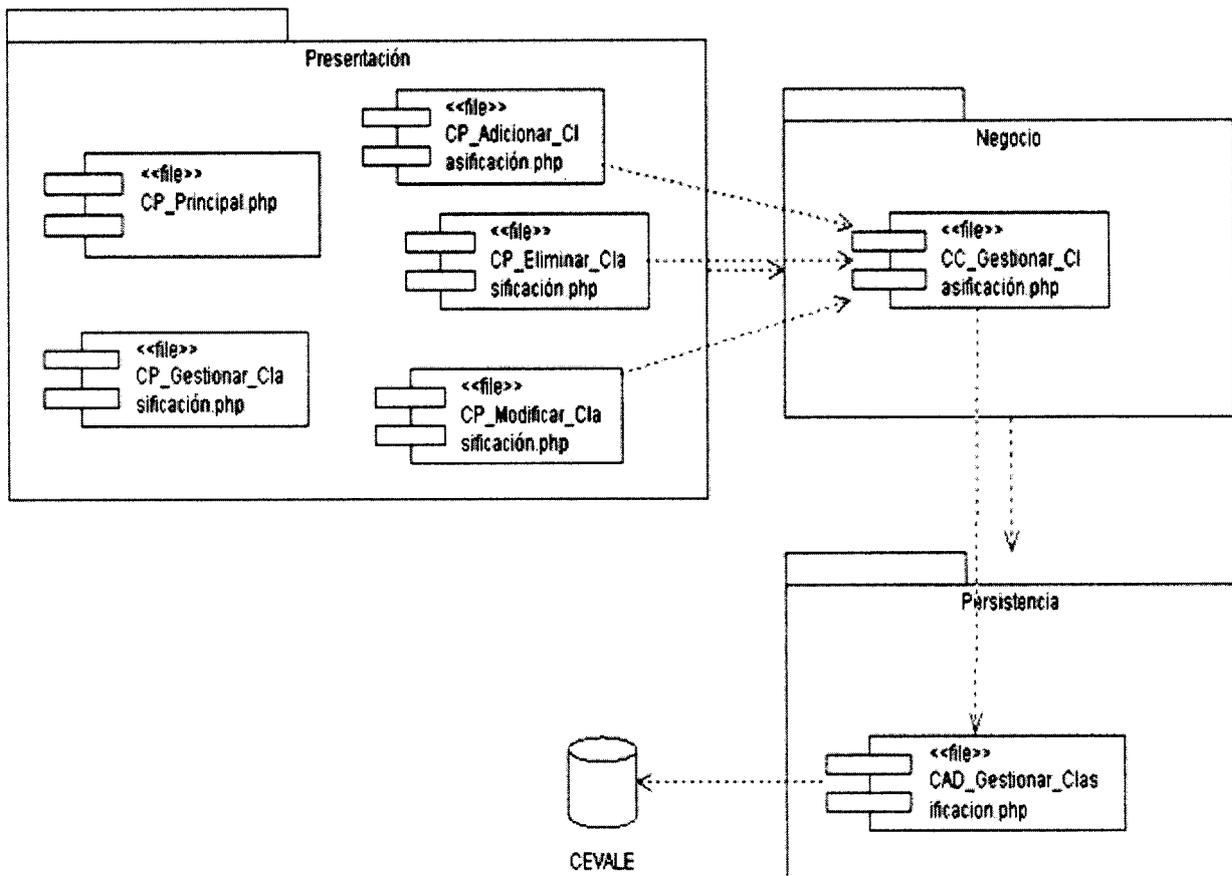


Figura 17: Diagrama de Componentes del CU_Gestionar_Clasificación

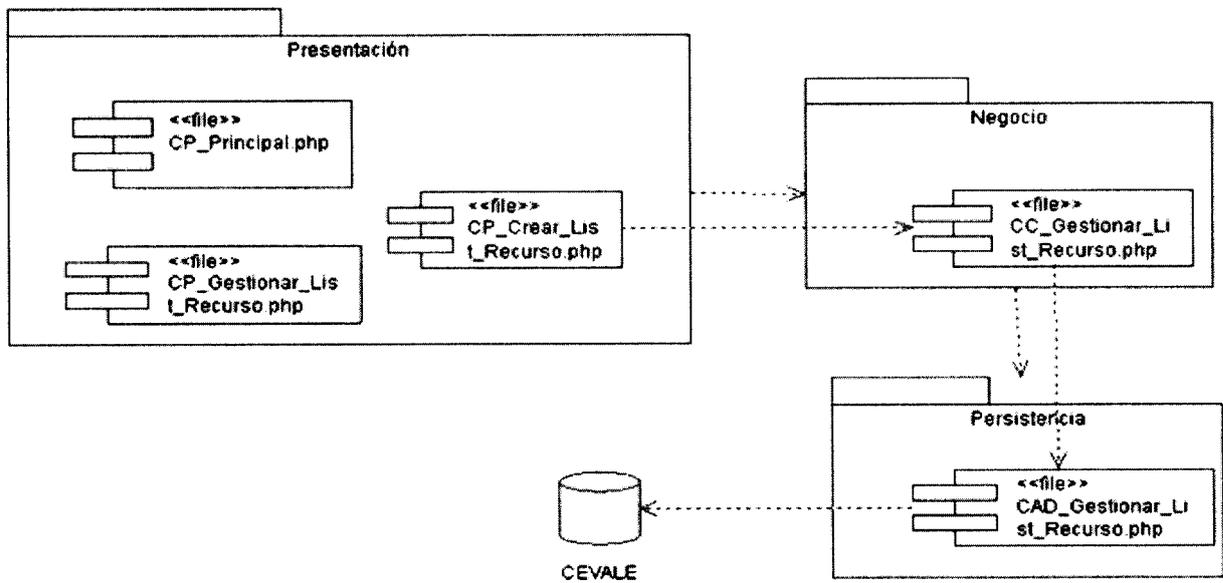


Figura 18: Diagrama de Componentes del CU_Gestionar_List_Recurso

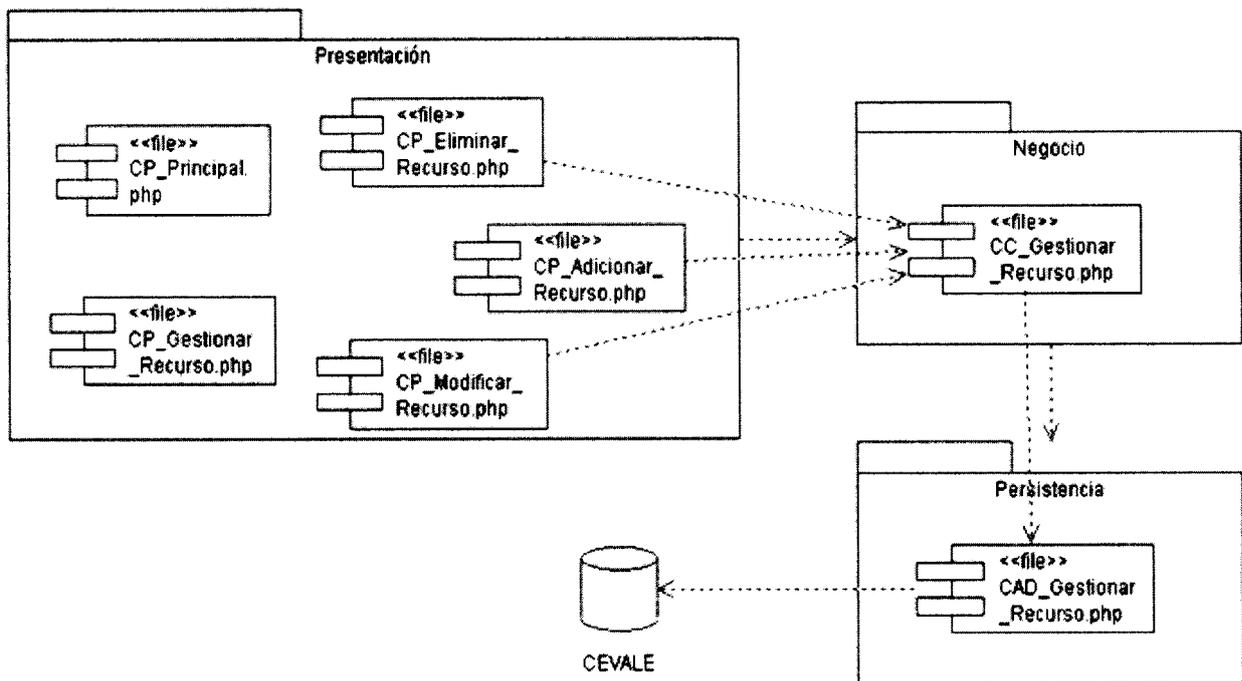


Figura 19: Diagrama de Componentes del CU_Gestionar_Recurso

4.7 Conclusiones parciales

En este capítulo, se representan las diferentes partes de la solución propuesta a través de diferentes diagramas utilizando las extensiones de UML para modelar los elementos relativos a la aplicación. Se realizaron los diagramas de clases de análisis y diseño donde se representaron las clases y sus asociaciones. Se definió además, el diseño de la base de datos, creando para ello el diagrama de clases persistentes, y a través de éste, el modelo de datos. También se confeccionó el diagrama de despliegue y se plantearon los principios de diseño de la aplicación Web.

CONCLUSIONES

En este trabajo se cumple exitosamente el objetivo trazado, realizándose el diseño del módulo de recursos cumpliendo con los requisitos solicitados por el cliente y a la altura de lo que se quería y necesitaba.

Los resultados de este trabajo contribuyen al autoaprendizaje de idiomas extranjeros en los CASIE de la UCI y serán de gran utilidad para la Universidad teniendo en cuenta que contribuye a mejorar el funcionamiento y la organización del proceso docente y educativo.

Los usuarios del sistema CEVALE, encontrarán un ambiente educativo variado, que los estimulará a desarrollar el proceso de autoaprendizaje de una forma más rápida y satisfactoria.

RECOMENDACIONES

Se recomienda para darle continuidad y mejorar la investigación desarrollada:

- Incorporar la base de datos de la UCI para las versiones posteriores.
- Agregarle a la base de datos otras clasificaciones de recursos.
- Realizar la implementación de la aplicación, para la solución real de los problemas existente en el módulo estudiado.
- Llevar la aplicación a otras plataformas libres, por ejemplo LINUX.

REFERENCIAS BIBLIOGRÁFICAS

1. *TICs para el desarrollo: un nuevo enfoque a partir de los Objetivos de Desarrollo del Milenio* Disponible en: <http://www.choike.org/nuevo/informes/2945.html>.
2. : MAITÉ GONZÁLES CRUZ, Y. M. C. *Cuba, las TICs y el Bloqueo*. 2007, Disponible en: <http://www.articulosgratis.com/content/view/491/277/>.
3. ARMENTEROS, Y. V. *LA TELEFORMACIÓN EN LA EDUCACIÓN SUPERIOR CUBANA. EXPERIENCIAS EN LA UCI*. 2005, Disponible en: <HTTP://WWW.VOZALMUNDO.COM/INDEX.PHP?ID=394>.
4. Disponible en: http://www.apac-eureka.org/revista/Volumen2/Numero_2_1/Pontes2005a.pdf.
5. BEATRIZ GARCÍA, M. G., GUSTAVO MORENO, IRENE DE OCHOA, NUVIAN RAMÍREZ, NORMA SEQUERA Y MARILENA ZUVIA. *FORMACIÓN DE DOCENTES EN EL USO DE RECURSOS DIDÁCTICOS PARA CONSTRUIR CONCEPTOS*. 2003, ISBN 13164910.
6. RIVERA., J. G. C. *LOS RECURSOS DIDÁCTICOS EN UN SISTEMA DE APRENDIZAJE AUTÓNOMO DE FORMACIÓN*.
7. YAPCI. *Definicion Joomla*. 2008, Disponible en: <http://www.yafaonline.com/porta1/joomla/joomla-noticias/definicion-joomla.html>.
8. *Acerca de e107*. Disponible en: <http://www.planetacms.com/e107/>.
9. PABLO LÓPEZ GARCÍA, M. L. S.-E. L. Disponible en: <http://profesores.universia.es/seccionEspecial.jsp?idEspecial=12&idSeccion=4230&title=CARACTERISTICAS-MOODLE>.
10. *Moodle*. Disponible en: <http://es.wikipedia.org/wiki/Moodle>.
11. *Metodología de desarrollo de software* Disponible en: http://www.wikilearning.com/curso_gratis/metodologias_de_desarrollo_de_software-metodologia_de_desarrollo_de_software/3617-1.
12. *Feature Driven Development*. Disponible en: www.fing.edu.uy/inco/cursos/gestsoft/Presentaciones/Procesos%20Agiles%20-%20G4/Procesos%20Agiles.doc.
13. VILLEGAS, A. A. *A propósito de programación extrema XP (eXtreme Programming)* Disponible en: <http://www.monografias.com/trabajos51/programacion-extrema/programacion-extrema.shtml>.

14. JUAN PABLO GOMEZ GALLEGO, J. G. *FUNDAMENTOS DE LA METODOLOGIA RUP RATIONAL UNIFIED PROCESS*. 2007, Disponible en: <http://www.scribd.com/doc/297224/RUP>.
15. MOLPECERES, A. *Procesos de desarrollo: RUP, XP Y FDD*. 2002,
16. *Lenguaje de modelado de objetos*. Disponible en: http://es.wikipedia.org/wiki/Lenguaje_de_modelado_de_objetos.
17. *Introducción a la Ingeniería de Software*. 2007,
18. *Unified Modeling Lenguaje*. Disponible en: <http://gidis.ing.unlpam.edu.ar/personas/glafuente/uml/uml.html>.
19. *Rational Rose*. 2005, Disponible en: http://searchcio-midmarket.techtarget.com/sDefinition/0,,sid183_gci516025,00.html.
20. *Rational Rose*. Disponible en: <http://www-306.ibm.com/software/awdtools/developer/rose/index.html>.
21. *Introducción a PHP 5*. Disponible en: <http://www.desarrolloweb.com/articulos/1696.php>.
22. *¿Qué es Eclipse?*. Disponible en: http://64.233.179.104/translate_c?hl=es&sl=en&u=http://www.eclipse.org/home/newcomers.php&prev=/search%3Fq%3Declipse%26hl%3Des.
23. *Eclipse (software)*. Disponible en: [http://es.wikipedia.org/wiki/Eclipse_\(software\)](http://es.wikipedia.org/wiki/Eclipse_(software)).
24. *Sistemas gestores de base de datos*. 2007, Disponible en: <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.
25. *Panorámica del sistema de gestión de base de datos MySQL*. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/what-is.html>.
26. *Las principales características de MySQL*. Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/features.html>.
27. *Mysql Comparado con otras Bases de Datos*. Disponible en: http://www.salnet.com.ar/inv_mysql/pag06_compara.htm.
28. CORNEJO, J. E. G. *Arquitectura en Capas ~ DNA Un camino hacia los procesos distribuidos*. 2001, Disponible en: www.docirs.cl/arquitectura_tres_capas.htm.

29. PEDRO PABLO ALARCÓN CAVERO, A. Y. P. *Técnicas avanzadas para el desarrollo de aplicaciones*. 2002, Disponible en: <http://www.oei.eui.upm.es/Asignaturas/BD/BD/docbd/tema/Arquitectura.pdf>
30. CORNEJO, J. E. G. *Arquitectura en Capas ~ DNA. Un camino hacia los procesos distribuidos*. 2001, Disponible en: www.docirs.cl/arquitectura_tres_capas.htm.
31. GUTIERREZ, J. A. S. *Patrones GRASP (patrones generales de software para asignar responsabilidades)* 2007, Disponible en: <http://jorgesaavedra.wordpress.com/2007/05/08/patrones-grasp-patrones-de-software-para-la-asignacion-general-de-responsabilidadparte-ii/>.
32. SAAVEDRA, J. *Patrones GRAPS* 2006, Disponible en: <http://jorgesaavedra.wordpress.com/2006/08/17/patrones-grasp-craig-larman/>.
33. *DIAGRAMAS DE CLASES*. 1999, Disponible en: <http://www.mcc.unam.mx/~cursos/Objetos/Cap8/cap8.html>.

BIBLIOGRAFÍA

1. B., C.R., Introducción a la Arquitectura de Software. 2006.
2. Daniel de la Cruz Heras, C. Z. R. Flash, PHP y MySQL. Contenidos dinámicos.
3. Gilfillan, I., La Biblia de MySQL
4. González, C. S. Aplicaciones en capas. 2004, Disponible en: <http://oness.sourceforge.net/proyecto/html/ch03502.html>.
5. Informáticos, R. S. Metodologías de Desarrollo de Software (MSD). 2005, Disponible en: <http://www.reynox.com.ar/sap/metodología.php>.
6. Ivar Jacobson, G. B., James Rumbaugh. El Proceso Unificado de Desarrollo de Software. 2000, ISBN 84-7829-036-2.
7. Larman, C., UML y Patrones. Introducción al análisis y diseño orientado a objetos. 2004.
8. Latina, O. ¿Por qué es importante UML? 2005, Disponible en: <http://www.osmosislatina.com/lenguajes/uml/basico.htm>.
9. L. Welling, L.T., Desarrollo Web con PHP y MySQL.
10. MySQL AB, MySQL 5.0. Reference Manual. 2007, Disponible en: <http://dev.mysql.com>.
11. Pressman, R. S. Ingeniería del Software. Un enfoque práctico. 5 ed. 2005.
12. Reyes, R.A.M., Los Patrones como un Medio del Diseño Orientado a Objetos.
13. Rolando Alfredo Hernández León, S. C. G. El Paradigma Cuantitativo Investigación Científica. 2002.
14. Saavedra, J.A., El mundo informático. Patrones GRASP 2006.

15. Sánchez, M. A. M. Metodologías de Desarrollo de Software. 2004, Disponible en: http://www.informatizate.net/articulos/metodologias_de_software_07062004.html.
16. Schamuller J., Aprendiendo UML en 24 horas. 2000. 448 p. ISBN 968-444-463-X.
17. Solmicro. ¿Qué significa y qué ventajas aporta la arquitectura en tres capas? 2007, Disponible en: http://www.solmicro.com/inicio.php?ID_CATEGORIA=25.
18. Vera, K. L. Ingeniería de Software-RUP-UML. 2006, Disponible en: <http://www.mmug.cl/articulos.php?id=287&tod=1>.
19. Wendy Boggs, M. B. UML White Rational Rose 2002.

ANEXOS

Anexo 1 Diagrama de Colaboración del CU Gestionar clasificación.

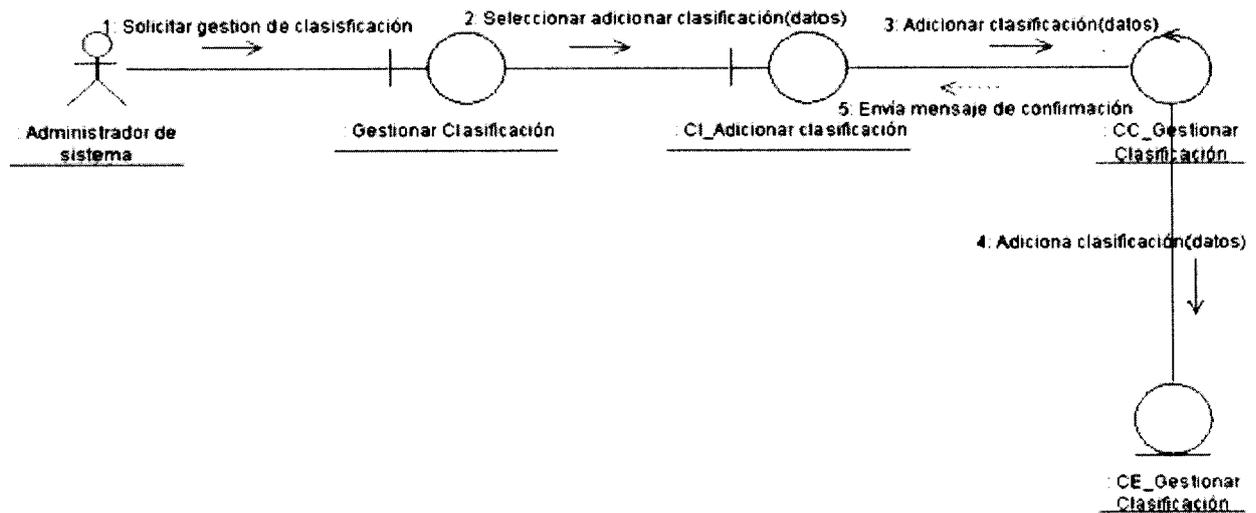


Figura 20: Diagrama de Colaboración del CU Gestionar clasificación. Adicionar clasificación

Anexo 2 Diagrama de Colaboración del CU Gestionar clasificación

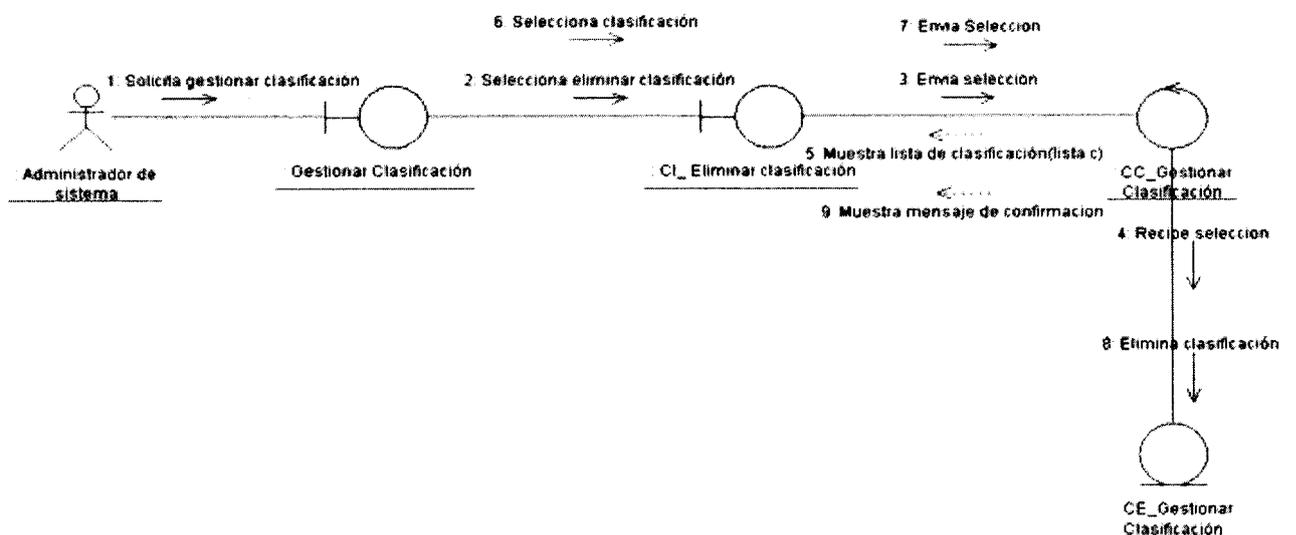


Figura 21: Diagrama de Colaboración del CU Gestionar clasificación. Eliminar clasificación

Anexo 3 Diagrama de Colaboración del CU Gestionar clasificación

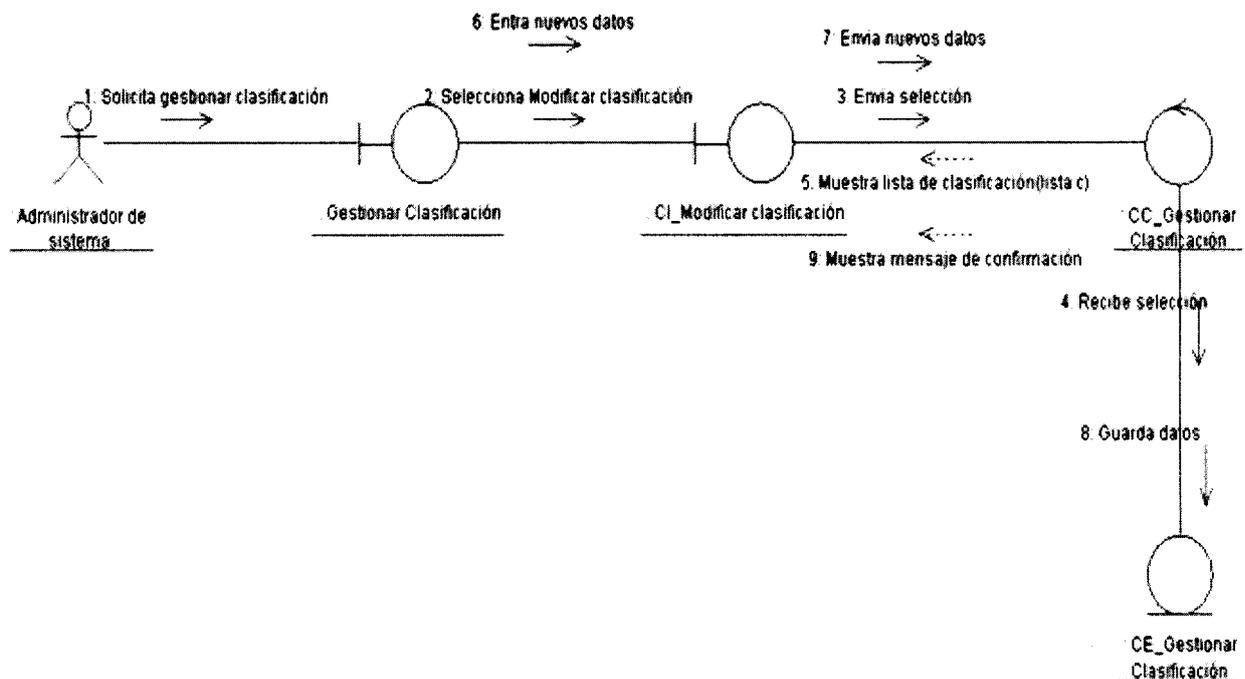


Figura 22: Diagrama de Colaboración del CU Gestionar clasificación. Modificar clasificación

Anexo 4 Diagrama de Colaboración del CU Gestionar recurso

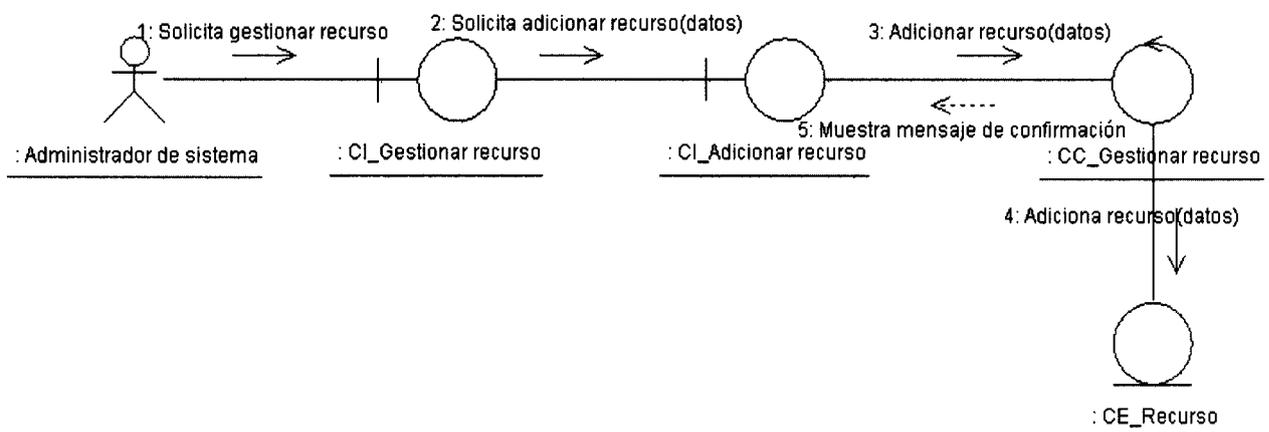


Figura 23: Diagrama de Colaboración del CU Gestionar recurso. Adicionar recurso

Anexo 5 Diagrama de Colaboración del CU Gestionar recurso

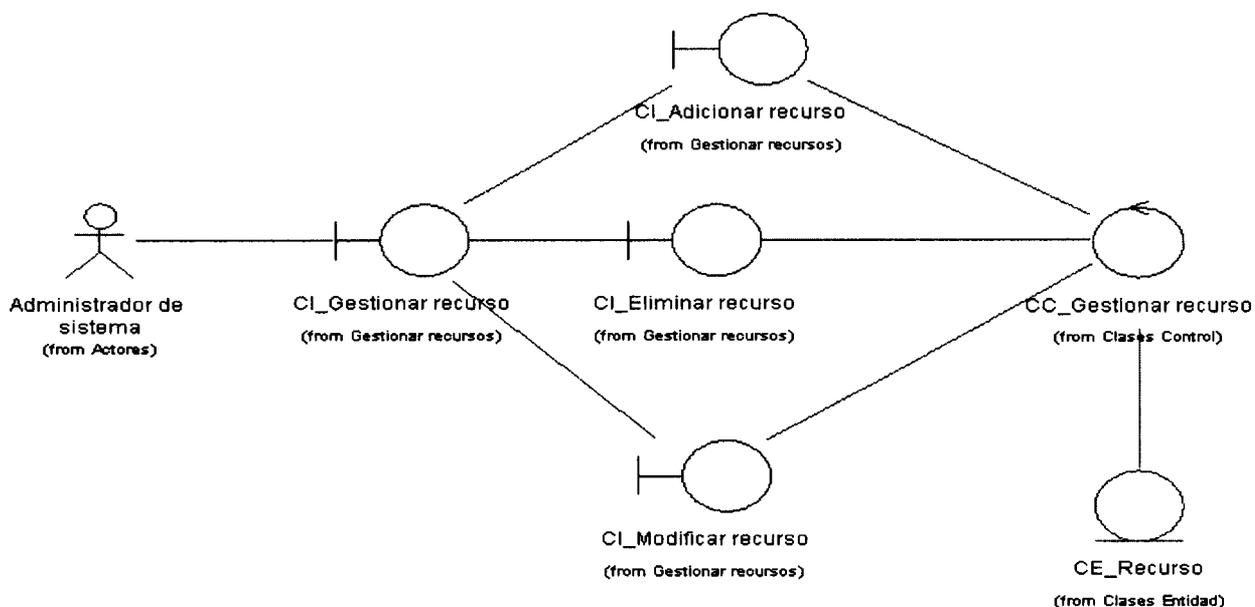


Figura 24: Diagrama de Colaboración del CU Gestionar recurso. Eliminar recurso

Anexo 6 Diagrama de Colaboración del CU Gestionar recurso

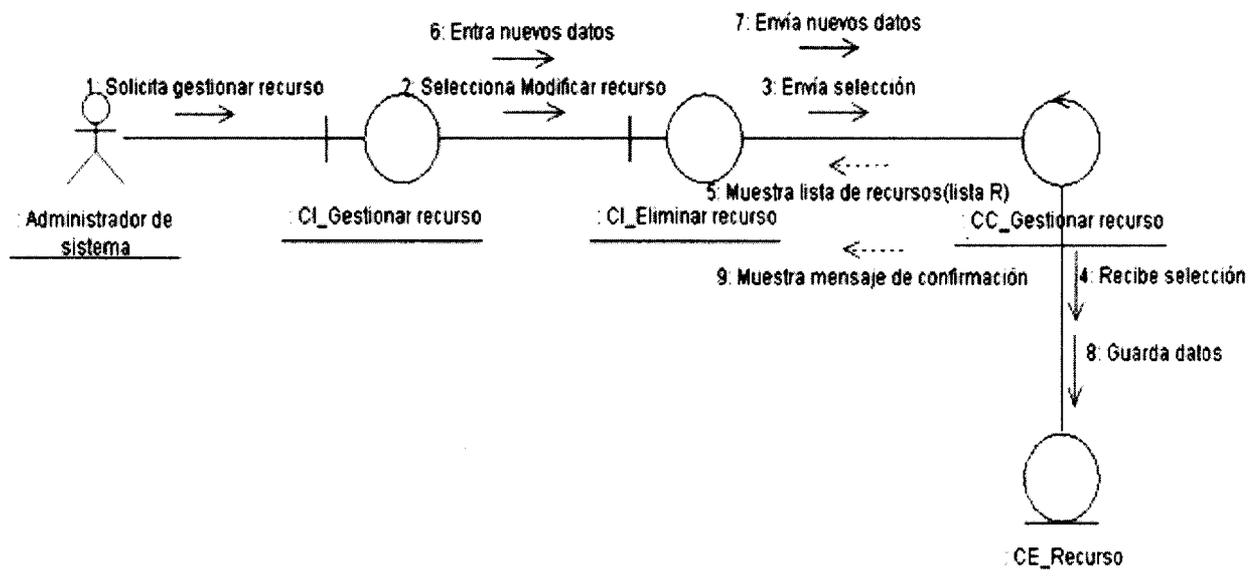


Figura 25: Diagrama de Colaboración del CU Gestionar recurso. Modificar recurso

Anexo 7 Diagrama de Colaboración del CU Gestionar listado de recursos

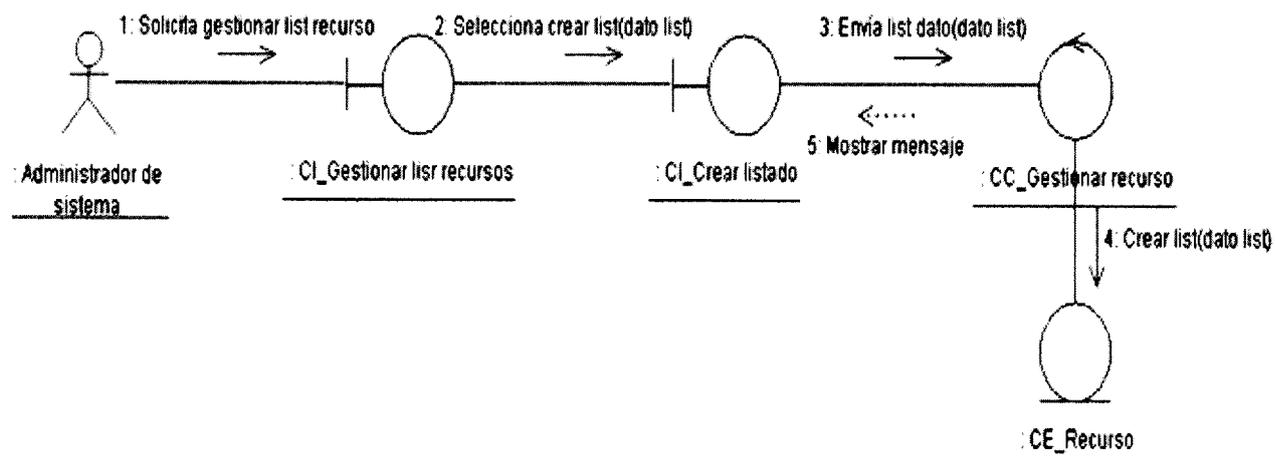


Figura 26: Diagrama de Colaboración del CU Gestionar listado de recursos. Crear lista de recursos

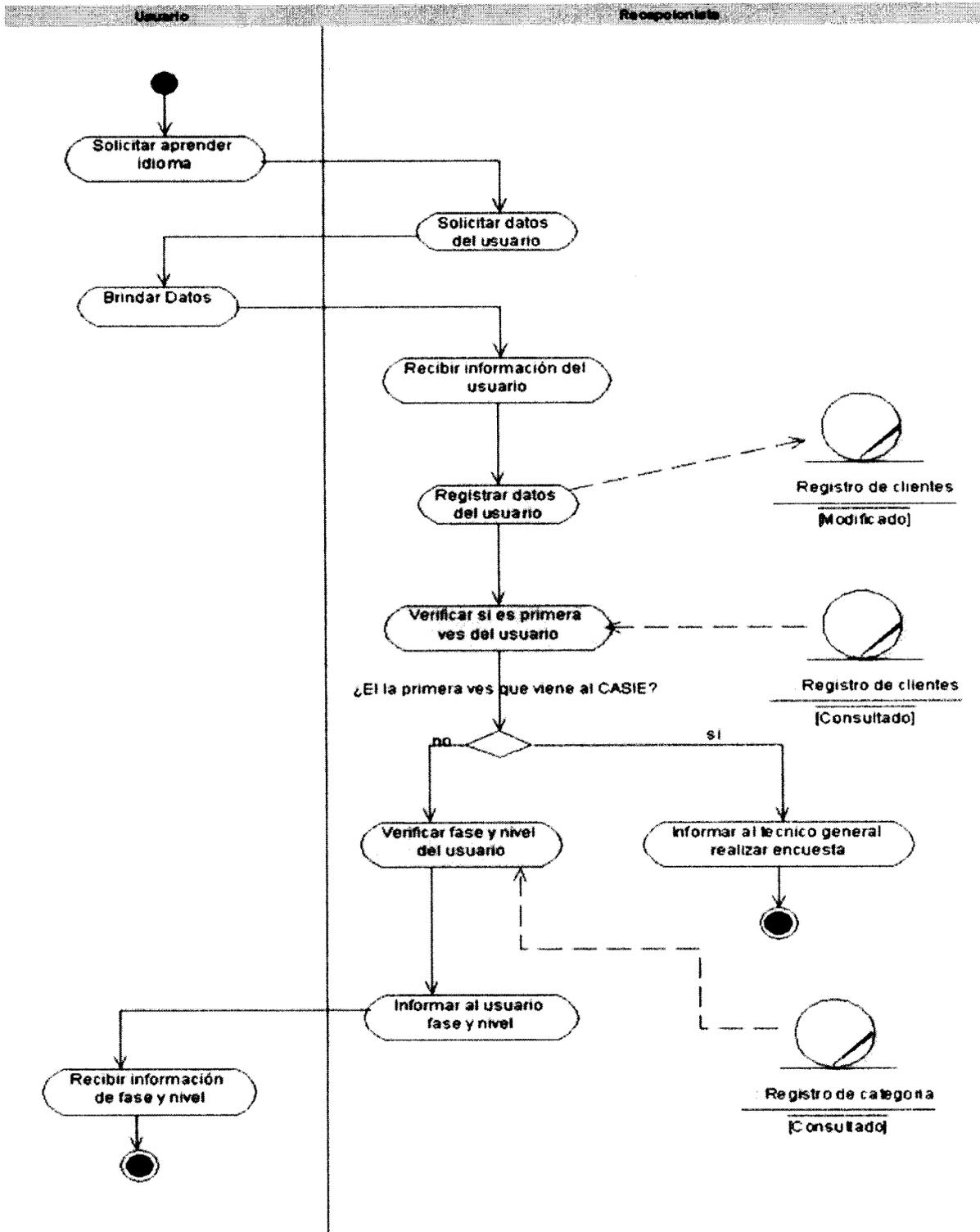


Figura 27: Diagrama de actividades del Caso de Uso Atender Usuario

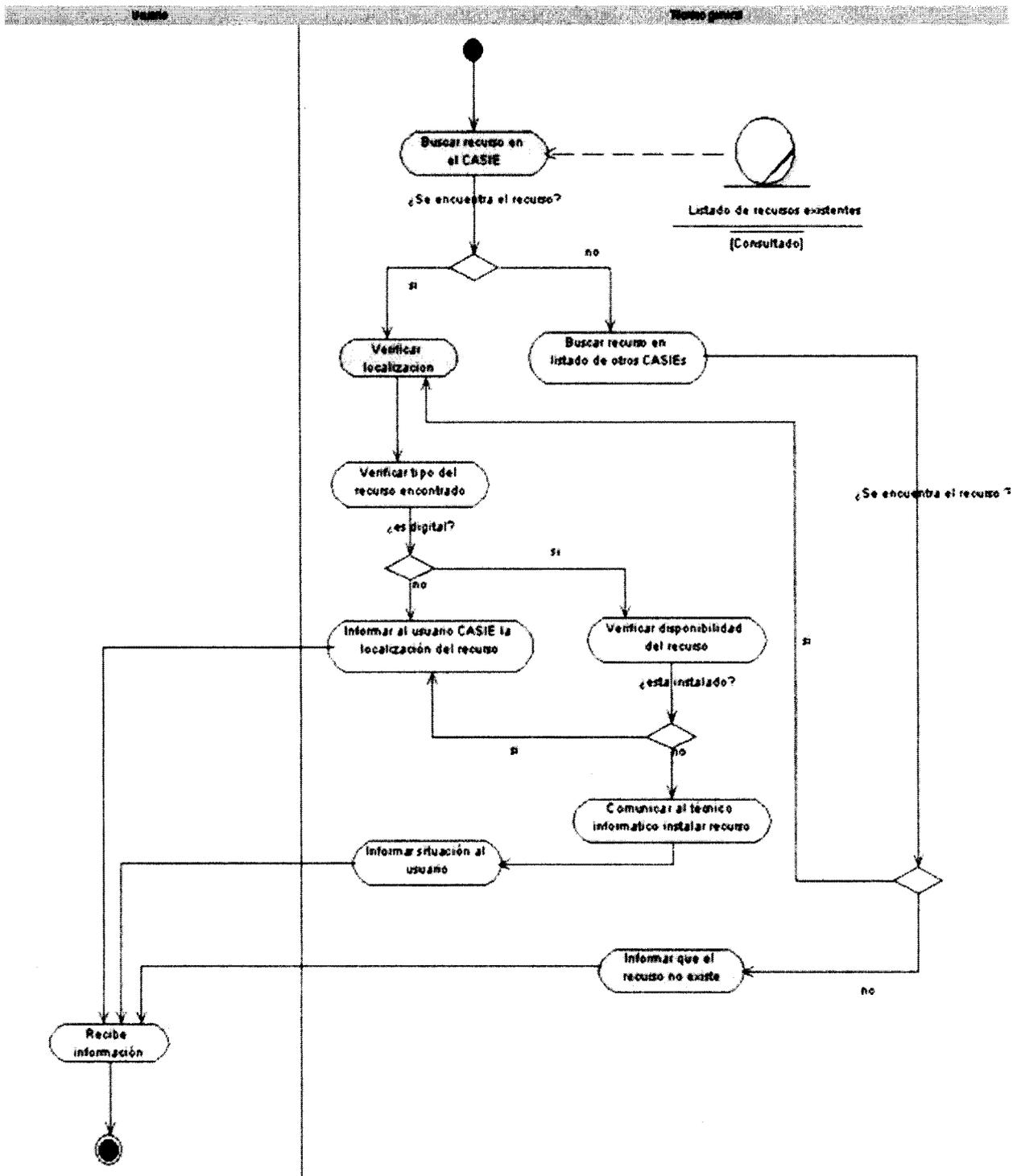


Figura 28: Diagrama de actividades del Caso de Uso Localizar Recurso

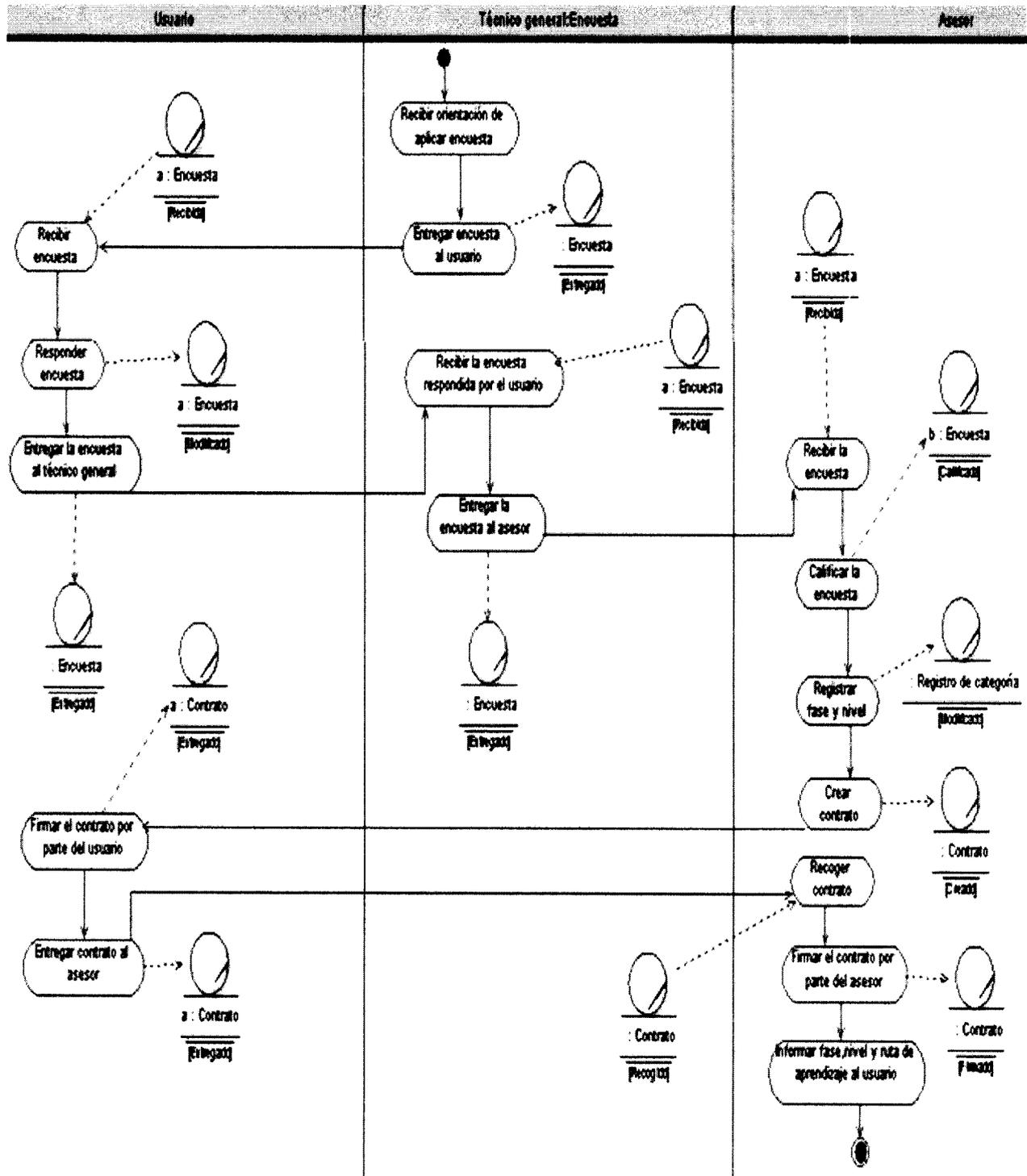


Figura 29: Diagrama de actividades del Casos de Uso Realizar encuesta

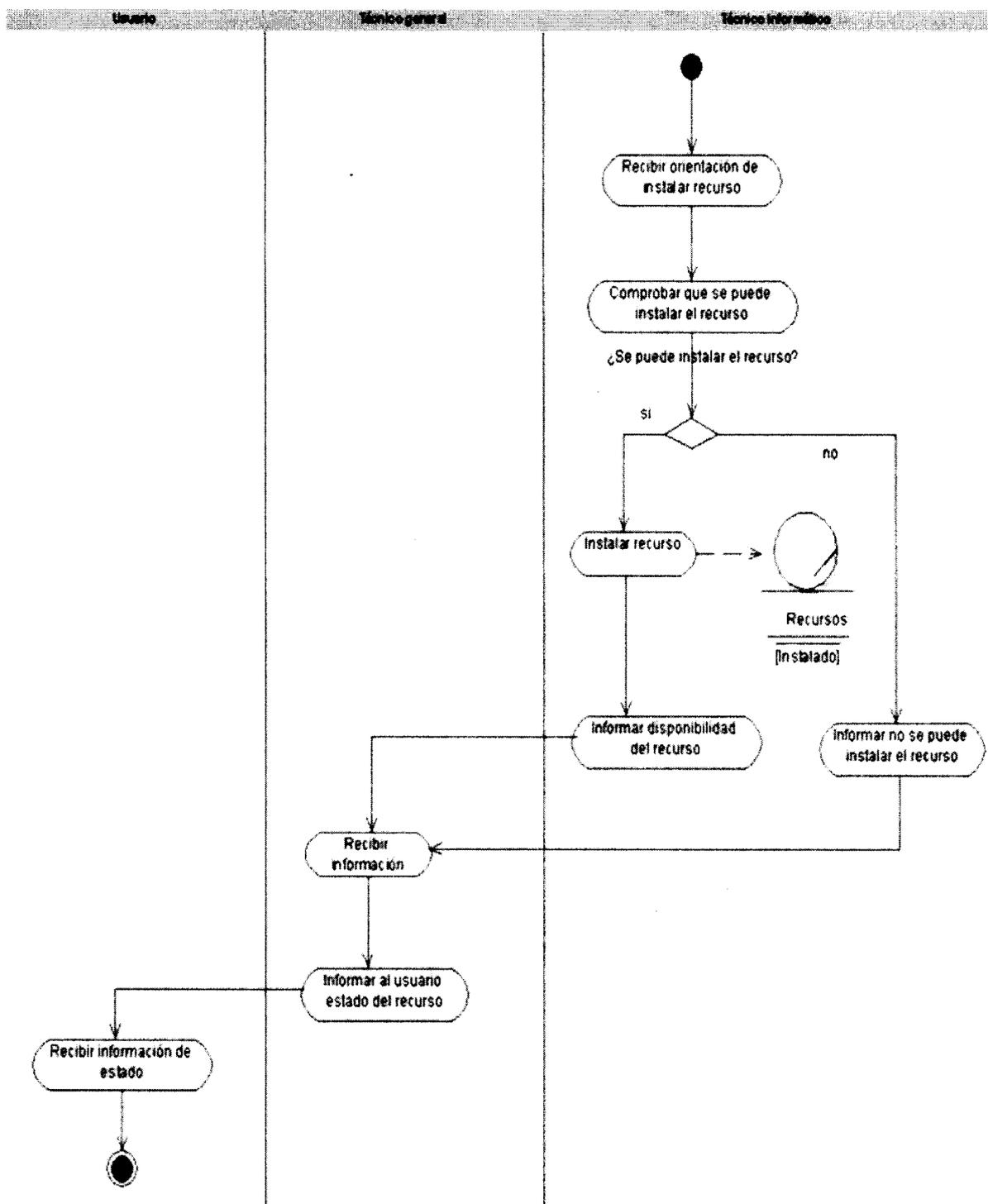


Figura 30: Diagrama de actividades del Casos de Uso Instalar Recurso

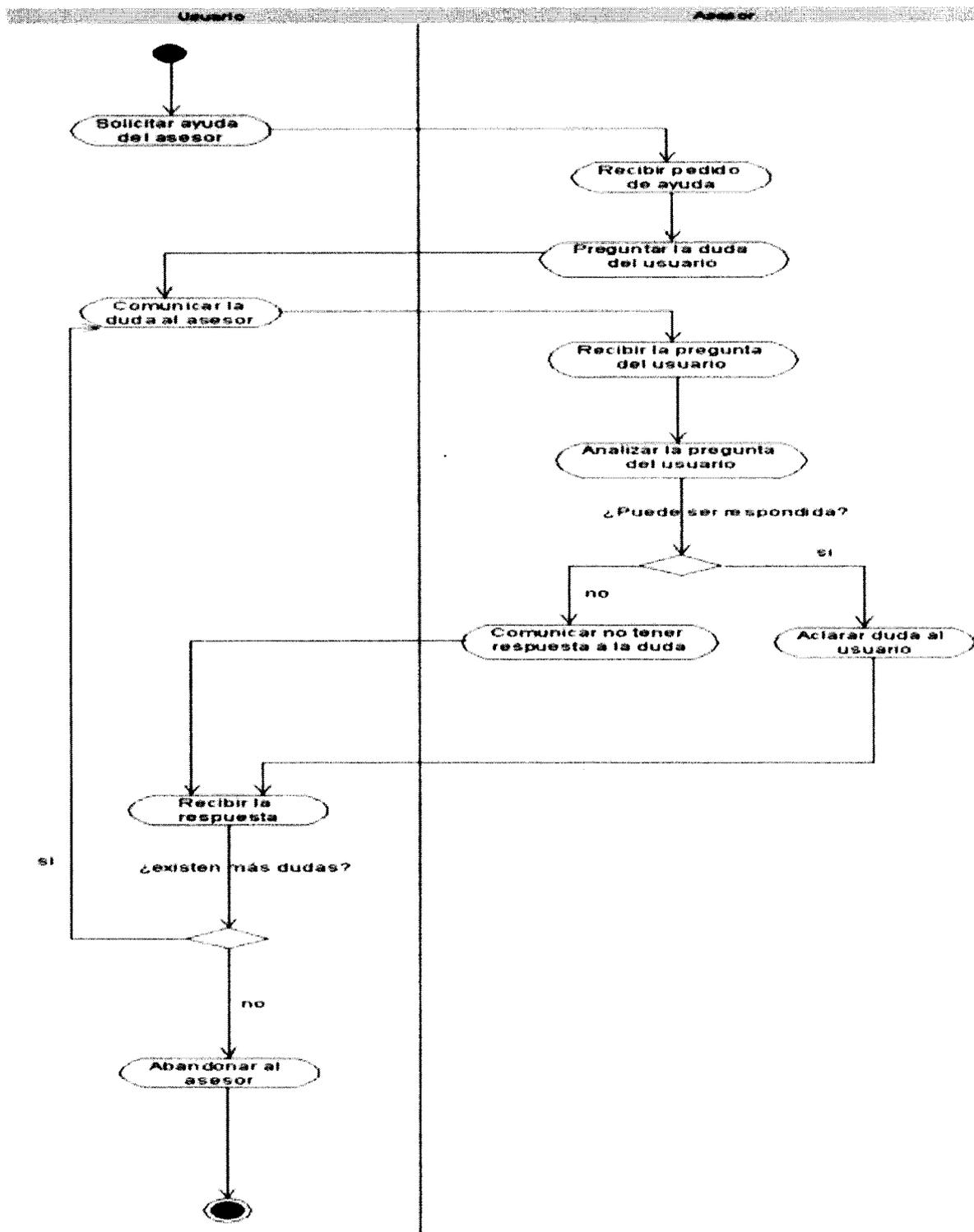


Figura 31: Diagrama de actividades del Caso de Uso Solicitar Asesoría

Proyecto CEVALE**Módulo de Recurso****MODELO DE ENTREVISTA**

Entrevistado: _____

Entrevistador: _____

Fecha: _____

Cuestionario

1. ¿Qué tipos de recursos existen actualmente en los CASIE?

Los usuarios que asisten a los CASIE pueden acceder a varios tipos de recursos entre los que se encuentran revistas, libros, multimedias, casetes de audio y de video.

2. ¿Los recursos como están organizados en los CASIE?

Se pueden encontrar en los CASIE recursos físicos y digitales. Los físicos se encuentran divididos por temas en estantes para un mejor control de éstos y los digitales, el usuario puede acceder a ellos a través de la red.

3. ¿Quién se encarga de controlar los recursos?

En los CASIE se encuentra un técnico general que es el encargado de localizar el recurso solicitado por el usuario y en caso de algún recurso digital que no esté disponible en la PC, el técnico general informa al técnico informático que es el encargado de instalarlo.

4. ¿En qué medidas son utilizados estos recursos?

Muchos son los usuarios que asisten al CASIE cada día en busca de recursos que le ayuden a un mejor aprendizaje en las lenguas extranjeras.

5. ¿Qué importancia tienen los recursos en el aprendizaje de lenguas extranjeras?

Los recursos le permiten al usuario trabajar de forma independiente en las dificultades que presente y convertir el estudio en un entretenimiento y salir de la monotonía de la forma de estudiar en las que están acostumbrados, apoyándolo así para lograr un eficiente autoaprendizaje.

GLOSARIO DE TÉRMINOS

A

Autoaprendizaje: Es el proceso al que se somete el usuario con el interés de aprender alguna cuestión teórica o técnica siguiendo una ruta de aprendizaje según su nivel y fase orientado o no por un asesor, con la conciencia de que deberá lograrlo poniendo su máximo empeño en ello y de que lo hará por sus propios medios, en tiempos que él decida.

C

Comportamiento indebido: Respuestas en el chat o foro dadas por un usuario que no esté dentro del trabajo realizado en el CASIE y su ruta de aprendizaje. Así como expresarse incorrectamente con el asesor o cualquier otro trabajador del CASIE.

Contrato: En el contrato se va a contener el espacio de tiempo que el usuario va a interactuar con CEVALE.

E

Esquema de asesoría: Es una organización por sesiones de trabajo de lunes a sábado que va a ver un asesor que se va a repetir por ejemplo: el lunes por la tarde, martes por la mañana, miércoles por la tarde, jueves por la mañana, viernes por la tarde, sábado por la mañana.

Expediente: Documento que va a contener el registro y contrato del usuario. El asesor debe tener a mano el Expediente del usuario para determinar su ruta de aprendizaje.

F

Fase: El usuario que haya realizado las dos encuestas, los ejercicios intermedios y los dos diagnósticos (inicial y final) con sus respectivos resultados recibirá por la plataforma en que fase de aprendizaje se encuentra y por donde debe empezar su real auto-aprendizaje. Las fases han de corresponderse con los semestres impartidos según el actual plan de Estudios. Serían 6 fases si se incluye el nivel 0 o Elemental: Fase 0, fase 1 (primer semestre), fase 2(segundo semestre), fase 3 (tercer semestre), fase 4 (IV semestre), fase 5 (V semestre).

N

Nivel: El usuario que haya realizado las dos encuestas, los ejercicios intermedios y los dos diagnósticos (inicial y final) con sus respectivos resultados recibirá por la plataforma un nivel determinado dependiendo del resultado obtenido, este puede ser: nivel básico, intermedio o avanzado dentro de cada fase.

R

Ruta de Autoaprendizaje: Se hace una prueba inicial para determinar si realmente el usuario está en el nivel que asegura y se proponen además una serie de ejercicios para darle la oportunidad de rectificar o recordar algo olvidado. Cada dato genera un perfil y cada perfil una ruta. Dichas rutas se comportan como un proceso cíclico por lo que deben permitir al usuario ir hacia delante o hacia atrás según vaya avanzando o no en el aprendizaje.

Registro de categoría: Archiva el resultado del usuario después de realizar el test.

I

Tiempo de asesoría: Tiempo de trabajo del que dispone el asesor, es decir, el asesor tiene ciertos días en la semana y dentro de estos, horarios de tiempos en el cual brindará se servicio.