



**Universidad de las Ciencias Informáticas**

**Facultad 7**

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

Aplicación informática para contribuir al desarrollo de la habilidad de comprensión lectora en el idioma Inglés, de los estudiantes del Departamento de Sistemas de Apoyo a la Salud

**Autores:** Odelys Muro Toledo  
Andrés Clavel Estupiñan

**Tutores:** MSc. Annia Arencibia Morales  
Lic. Meybel Rivero Rivadeneira

**La Habana, junio de 2013**  
**“Año 55 de la Revolución”**

## **Datos de Contacto**

### **Tutores:**

- **MSc. Annia Arencibia Morales** (aarencibia@uci.cu): Graduada de Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas en el año 2007. Profesora de la Universidad de las Ciencias Informáticas, pertenece al Centro de Informática Médica (CESIM), posee categoría docente de profesor Asistente. Máster en Informática Aplicada. Profesora de Metodología de la Investigación Científica, e imparte postgrado de Ingeniería de Software. Pertenece al Departamento de Sistemas de Apoyo a la Salud.
- **Lic. Meybel Rivero Rivadeneira** (meybel@uci.cu): Graduado de Licenciada de Lengua Inglesa, en el Pedagógico: Capitán Silverio Blanco Núñez de Sancti-Spiritus en el año 2004. Profesor de la Universidad de las Ciencias Informáticas, pertenece al Departamento de Ciencias Sociales y Humanidades de la Facultad 7. Imparte la asignatura de Inglés V y se encuentra colaborando en el Departamento de Sistemas de Apoyo a la Salud del Centro de Informática Médica (CESIM). Se encuentra cursando la tercera edición de la siguiente Maestría: Las Tecnologías en los procesos Educativos (TIC) en la CUJAE.

## Agradecimientos

*En primer lugar quiero hacer los agradecimientos más grandes y especiales a mis padres por traerme a este mundo, por hacerme la mujer que soy, por todo lo que han hecho para que yo pudiera hacer realidad este sueño que era graduarme y tener mi título de ingeniera.*

*A mis tutoras Annia y Meybel por sus consejos, paciencia y opiniones, por haberme ayudado y apoyado en todo el desarrollo de la tesis.*

*A mi familia por apoyarme en todo, por ser siempre atentos conmigo, por haberme cuidado y aconsejado en todo momento, a mis tíos, especialmente mi tía Margarita que ha sido como una madre para mí, a mis primos que más que mis primos son mis hermanos, a mi hermano que fue mi profesor y aprendí muchísimo con su ejemplo y guía, a mi madre y padre que los amo, a mis abuelos, los que están y los que no se encuentran entre nosotros por cosas de la vida, los quiero mucho.*

*Agradecer a mi compañero de tesis por haber compartido conmigo este sueño y ser el mejor compañero de tesis.*

*A mi novio que lo amo mucho, por estar a mi lado en los buenos y malos días que he tenido. A mis suegros por quererme como una hija.*

*A todos mis compañeros de grupo, los que están ahora y los que estuvieron conmigo desde primer año, a mis compañeras de apartamento, las que están y las que estuvieron, a Eddialis por ser una gran amiga, Annabel, Dora, Leskenia, Daimy; en fin gracias a todas, nunca las olvido ya que compartimos muchas cosas juntas todos estos años.*

*A las muchachitas del grupo que me han ayudado mucho estos tres años que llevamos juntas, especialmente Arlety por ayudarme tanto en la tesis, a Claudia y su novio, a Zuleidy, Yeni, Yaima, en fin a todas gracias por ayudarme y apoyarme en todo, por ser tan buenas amigas.*

*A todos mis amigos que compartieron momentos conmigo, a los del pre que continuaron acompañándome en la universidad, que nunca me han dejado a un lado y siempre han estado ahí atentos.*

*A mis profesores por enseñarme y apoyarme estos cinco años de carrera.*

*Agradecerle a mis vecinos Cito y Alina, por ser tan buenas personas y quererme como una sobrina.*

*Agradecerles a todos los que de una forma u otra me ayudaron a cumplir este sueño.*

*Gracias a la UCI por este sueño de ser hoy una profesional.*

Odelys

## *Agradecimientos*

---

*Le agradezco a todas aquellas personas que de una forma u otra hicieron posible que yo haya logrado llegar hasta aquí, a mis tutoras por haberme apoyado todo este curso en la tesis, le agradezco a mis compañeros de grupo que me han ayudado durante toda la carrera, a mis amigos del barrio donde vivo.*

*A mi compañera de tesis por haber sido mucho mejor de lo que yo esperaba y por todo su esfuerzo y su abnegación en la realización de la tesis durante todo este curso.*

*A toda mi familia por todo lo que ha hecho por mí, a mi novia por todo el amor y el cariño que recibo de ella, y le agradezco especialmente a mi padrastro que ha sido un verdadero padre conmigo, a mis hermanos que me han dado todo lo mejor de ellos y por ser los mejores hermanos que una persona pueda tener; y por último le agradezco eternamente a mi mamá, por ser la mejor madre del mundo, por haberme dado todo el apoyo que necesité durante toda mi vida, por haberme dado la educación que tengo, por hacer de mi la persona que soy, y por todas las cosas que ha hecho por mí, que realmente no tengo palabras para describirlo.*

*Andrés*

### Dedicatoria

*Este trabajo de diploma lo quiero dedicar a las dos personas que más amo en este mundo, a mi mamá Odalys y a mi papá Carlos, porque sé que los hago muy feliz con este logro. Por ser los mejores padres del mundo, padres que han luchado incansablemente conmigo desde el principio, que me han apoyado en todo lo bueno y malo que he pasado en mi vida, quitándose ellos para poder darme a mí. Espero enormemente que estén orgullosos de su hija y que sientan la alegría que les regalo, esto no es solo un triunfo mío, sino de los tres ya que sin ellos este momento no hubiese sido posible.*

*Odelys*

*A mi mamá, por ser mayor merecedora de este logro que yo, ya que ella es el motivo por el cual he llegado hasta aquí.*

*Andrés*

### Resumen

Actualmente la relevancia del aprendizaje del idioma Inglés es reconocida en todos los ámbitos. Este es una herramienta de comunicación global y una vía de acceso a mayores conocimientos. Lograr que todos los graduados sean capaces de comunicarse usando las habilidades de la lengua inglesa, es uno de los retos que tiene la Universidad de las Ciencias Informáticas. Los estudiantes de dicha universidad reciben durante los cuatro primeros años de la carrera los conocimientos básicos del idioma Inglés; pero se ha comprobado que al incorporarse a los proyectos productivos del Departamento de Sistemas de Apoyo a la Salud (SAS) presentan dificultad en la habilidad de comprensión lectora; específicamente para consultar bibliografía especializada y actualizada referente a temas productivos.

Se desarrolló una aplicación informática llamada: “Sistema de Aprendizaje para el idioma Inglés” (SAII), que les facilitará a los estudiantes del Departamento de Sistemas de Apoyo a la Salud, realizar actividades de temas productivos. Para guiar el desarrollo de la aplicación, se utilizó la Metodología de Desarrollo de Software, Proceso Unificado de Desarrollo y Enterprise Architect como herramienta de modelado haciendo uso del Lenguaje Unificado de Modelado. C Sharp y XML como lenguajes. SQLite y PostgreSQL como sistemas gestores de base de datos, como herramientas de administración de base de datos SQLiteManager y pgAdmin, como herramienta de desarrollo, MonoDevelop. La aplicación puede utilizarse desde cualquier estación de trabajo, y le permite al profesor crear ejercicios diferenciados por estudiante.

**Palabras clave:** *aplicación informática, habilidades, comprensión lectora*

## Índice

|   |    |
|---|----|
| Introducción .....  | 4  |
| Capítulo 1: Fundamentación teórica de la investigación.....   | 9  |
| 1.1 Antecedentes.....   | 9  |
| 1.1.1 Sistemas existentes en el ámbito internacional.....   | 9  |
| 1.1.2 Sistemas existentes en el ámbito nacional.....  | 10 |
| 1.2 Tecnologías, metodologías y herramientas.....   | 12 |
| 1.2.1 C Sharp v4.0.....   | 12 |
| 1.2.2 JAVA .....  | 13 |
| 1.2.3 XML v1.0 .....  | 13 |
| 1.2.4 UML v2.1 .....  | 13 |
| 1.2.5 Proceso Unificado de Desarrollo.....  | 14 |
| 1.2.6 Arquitectura en Tres Capas .....  | 14 |
| 1.2.7 Modelo Vista Controlador .....  | 14 |
| 1.2.8 MonoDevelop v3.0.....   | 15 |
| 1.2.9 Netbeans IDE v6.9.....  | 16 |
| 1.2.10 Enterprise Architect v7.1 .....  | 16 |
| 1.2.11 SQLiteManager v0.7 .....   | 16 |
| 1.2.12 pgAdmin v1.6.....  | 17 |
| 1.2.13 Oracle Berkeley DB (BDB).....  | 17 |
| 1.2.14 MongoDB.....   | 17 |
| 1.2.15 SQLite v3.7.....   | 18 |
| 1.2.16 PostgreSQL v9.2 .....  | 18 |
| 1.3 Herramientas, lenguajes, metodología y tecnologías propuestos para la solución del problema planteado ..... | 19 |
| Capítulo 2: Propuesta y características del sistema SAII .....  | 23 |
| 2.1 Descripción de la aplicación propuesta .....  | 23 |
| 2.2 Modelo del dominio.....   | 26 |
| 2.2.1 Conceptos fundamentales .....   | 27 |
| 2.2.2 Diagrama del modelo de dominio.....   | 28 |
| 2.3 Especificación de los requisitos de software .....  | 28 |
| 2.3.1 Requisitos Funcionales (RF).....  | 29 |

---

|                            |  |    |
|----------------------------|--|----|
| 2.3.2                      | Requisitos No Funcionales (RNF).....                     | 30 |
| 2.4                        | Modelo de casos de uso del sistema .....                 | 33 |
| 2.4.1                      | Definición de los actores.....                           | 33 |
| 2.4.2                      | Diagrama de casos de uso del sistema .....               | 33 |
| 2.4.3                      | Descripción textual de los caso de uso del sistema ..... | 34 |
| Capítulo 3:                | Análisis, diseño e implementación del sistema SAII.....  | 36 |
| 3.1                        | Descripción de la arquitectura.....                      | 36 |
| 3.2                        | Patrones de Diseño .....                                 | 36 |
| 3.3                        | Análisis .....   | 37 |
| 3.3.1                      | Modelo de análisis .....                                 | 37 |
| 3.3.1.1                    | Diagramas de clases del análisis.....                    | 38 |
| 3.3.1.2                    | Diagramas de interacción .....                           | 39 |
| 3.2                        | Modelo de diseño .....                                   | 42 |
| 3.2.1                      | Diagramas de clases del diseño .....                     | 42 |
| 3.2.2                      | Modelo de datos .....                                    | 50 |
| 3.3                        | Modelo de implementación .....                           | 52 |
| 3.3.1                      | Diagrama de componentes.....                             | 52 |
| 3.3.2                      | Diagrama de despliegue .....                             | 54 |
| 3.4                        | Seguridad .....  | 56 |
| Conclusiones               | .....  | 57 |
| Recomendaciones            | .....  | 58 |
| Referencias bibliográficas | .....  | 59 |
| Bibliografía.....          | .....  | 63 |
| Anexos.....                | .....  | 71 |



### **Introducción**

Las Tecnologías de la Información y las Comunicaciones (TIC) cumplen un papel determinante en la sociedad del conocimiento, especialmente en aquellas naciones que van a la vanguardia del desarrollo. (Nuryr, 2008). Las TIC han permitido globalizar la comunicación, facilitando la interconexión entre las personas e instituciones y eliminando barreras espaciales y temporales. (Rivas, y otros, 2008 pág. 4)

Las TIC, en Cuba, se definen como el proceso de utilización ordenada y masiva de las tecnologías, para satisfacer las necesidades de información y conocimiento de todas las personas y esferas de la sociedad (Ferrara, 2010). La esfera de la educación no se encuentra ajena a este fenómeno, pues ha sido preocupación constante de todas las personas involucradas con este sector, utilizar estas para el desarrollo del proceso docente educativo. Las TIC se han puesto en práctica en disímiles materias, siendo uno de ellas el idioma Inglés (idioma universal); el estudio de este en Cuba se ha insertado desde edades muy tempranas, comenzando en los primeros niveles educacionales.

En la enseñanza primaria se inicia el estudio en cuarto grado, a partir del uso cotidiano de mensajes orales y escritos sencillos, que permiten conformar el vocabulario básico para comunicarse. Luego en la secundaria básica, se imparte una materia más avanzada que amplía los horizontes de aprendizaje; en el pre universitario, se desarrolla con mayor profundidad estos conocimientos y se potencian de una manera más técnica en la universidad. En el contexto de Cuba, el idioma Inglés es definido como una lengua extranjera, porque no se emplea como lengua de comunicación entre sus habitantes y los estudiantes no tienen acceso inmediato a su uso fuera de la sala de clases. (Brown, 2007)

En las universidades cubanas es de vital importancia el uso de este idioma, dentro de los retos que presentan, está lograr que todos los graduados sean capaces de comunicarse usando las habilidades de la lengua inglesa (Márquez, 2007 p. 3). La Universidad de las Ciencias Informáticas (UCI), por ser líder del proceso de informatización de la sociedad, también presenta este reto, debido al amplio intercambio comercial tras convenios con países amigos, interesados por los productos informáticos que se desarrollan en la misma. Aprender Inglés en la UCI, se hace cada día más relevante, pues la carrera de informática requiere de estar en constante actualización tecnológica.

Dentro del plan de estudio de la carrera de Ingeniero en Ciencias Informáticas, se imparten cinco asignaturas de lengua inglesa, desde segundo hasta cuarto año. Inglés I y II se enseñan con fines comunicativos e Inglés III, IV y V con fines específicos (Inglés técnico o de la especialidad). A continuación se presenta una tabla especificando la distribución de las asignaturas anteriormente mencionadas, teniendo en cuenta el año y el semestre en que se imparten así como las habilidades que se desarrollan en cada una de ellas:

Tabla 1. Distribución de las asignaturas del idioma Inglés

| <b>Asignaturas</b> | <b>Año</b> | <b>Semestre</b> | <b>Habilidades que se desarrollan</b>                    |
|--------------------|------------|-----------------|--|
| Inglés I           | Segundo    | Primero         | Comprensión auditiva y lectora, expresión oral y escrita |
| Inglés II          | Segundo    | Segundo         |  |
| Inglés III         | Tercero    | Primero         |  |
| Inglés IV          | Tercero    | Segundo         |  |
| Inglés V           | Cuarto     | Primero         |  |

Los estudiantes al llegar al tercer año de la carrera se especializan en un perfil orientado a la informatización de la esfera social y económica de Cuba, dentro de un centro productivo de la UCI. El Centro de Informática Médica (CESIM), está compuesto por varios departamentos, siendo uno de estos el Departamento de Sistemas de Apoyo a la Salud (SAS). Al incorporarse los estudiantes a los proyectos de dicho departamento, le son asignadas tareas, enfrentándose a algunos problemas (consulta de bibliografía en idioma Inglés) que se enuncian a continuación:

- Los estudiantes al llegar a la producción necesitan consultar bibliografía especializada y actualizada en idioma Inglés (respecto a tecnologías, herramientas, metodologías, etc.), presentando dificultad en las habilidades de: comprensión auditiva y lectora, expresión oral y escrita. Debido a esto, realizan búsquedas de bibliografía en español, cuando las más completas se encuentran en Inglés (por ser sus creadores de países de habla inglesa).
- Los estudiantes no pueden comunicarse a plenitud con especialistas extranjeros que visitan la UCI, ya sea en eventos, conferencias u otros espacios, para brindar información referente a los temas que se desarrollan en los proyectos productivos. Debido a que presentan dificultad en las habilidades del idioma Inglés.

- Los estudiantes presentan dependencia de herramientas (como traductores y diccionarios) cuando al realizar alguna tarea en la producción requieren consultar bibliografía en idioma Inglés.

Se aplicó una encuesta a los estudiantes del Departamento SAS (ver Anexo 1), para detectar otras dificultades relacionadas con el idioma Inglés. La mayoría de los problemas detectados coinciden con los descritos anteriormente, excepto de que existe desconocimiento de herramientas informáticas para el estudio de temas productivos en idioma Inglés (ver Anexo 2). A través de la encuesta, se midió el grado de satisfacción de los estudiantes por el estudio del idioma Inglés, haciendo uso de la técnica de ladov (ver Anexo 3), demostrando insatisfacción por parte de estos.

De acuerdo a lo planteado anteriormente, surge como **problema a resolver**: ¿Cómo contribuir al desarrollo de la habilidad de comprensión lectora en el idioma Inglés, de los estudiantes del Departamento de Sistemas de Apoyo a la Salud? Enmarcado en el **objeto de estudio**: proceso de desarrollo de habilidades del idioma Inglés, haciendo uso de una aplicación informática; centrándose en el **campo de acción**: desarrollo de la habilidad de comprensión lectora en el idioma Inglés, de los estudiantes que se incorporan a los proyectos productivos del Departamento de Sistemas de Apoyo a la Salud, del Centro de Informática Médica. Para dar solución al problema anteriormente planteado se define como **objetivo general**: desarrollar una aplicación informática, que les permita a los estudiantes que se incorporan a los proyectos productivos del Departamento de Sistemas de Apoyo a la Salud, mejorar la habilidad de comprensión lectora en el idioma Inglés.

Para dar solución al objetivo planteado se definieron las siguientes **tareas de la investigación**:

- Realizar el análisis de sistemas internacionales y nacionales utilizados para el aprendizaje del idioma Inglés, estableciendo similitudes con la investigación en curso.
- Proponer una aplicación informática, que le permita a los estudiantes del Departamento de Sistemas de Apoyo a la Salud, mejorar la habilidad de comprensión lectora en el idioma Inglés.
- Desarrollar los artefactos correspondientes a los flujos de trabajo propuestos por la metodología de desarrollo, Proceso Unificado de Desarrollo (RUP), sirviendo de base en la implementación de la aplicación informática propuesta.
- Definir herramientas y tecnologías para el desarrollo de la aplicación informática propuesta.

- Implementar la aplicación informática propuesta utilizando las herramientas y tecnologías definidas.

Los principales **métodos de investigación** empleados son:

### **Métodos teóricos:**

- **Método Histórico-Lógico:** se analizó la forma en que se aprende el idioma Inglés haciendo uso de aplicaciones informáticas, identificándose los aspectos positivos y negativos.
- **Método Analítico-Sintético:** se utiliza para lograr extraer los elementos esenciales de la bibliografía consultada, sintetizando la necesidad del estudio del idioma Inglés desde la producción.
- **Método Modelación:** se utiliza para explicar la realidad con la creación de diagramas. Mediante su utilización se elaboraron diferentes diagramas que brindaron información clara sobre el tema de estudio.

### **Métodos Empíricos:**

- **Encuesta:** se realizó una encuesta estructurada a los estudiantes del Departamento SAS, recopilando información referente al grado de satisfacción que ellos presentan, respecto al estudio de los temas de la producción en idioma Inglés y además conocer los temas que prefieren estudiar.(Ver anexo 1)

Con el desarrollo de la aplicación se esperan los siguientes beneficios:

- El estudiante contará con una aplicación que presentará contenidos de temas productivos en idioma Inglés y esta podrá ser utilizada desde cualquier estación de trabajo.
- La herramienta permitirá al profesor crear actividades diferenciadas en dependencia del nivel de conocimiento de cada uno de sus estudiantes.
- Haciendo uso de la aplicación, se podrá incorporar el estudio del idioma Inglés como una tarea más de la producción; contribuyendo de esta forma, que los estudiantes del departamento SAS, mejoren la habilidad de comprensión lectora en este lengua.

El presente trabajo está dividido en tres capítulos que se describen a continuación:

## **Capítulo 1: Fundamentación teórica de la investigación**

En este capítulo se realiza un análisis de sistemas que permiten el desarrollo de las habilidades del idioma Inglés. Se describen y definen las tecnologías, metodologías y herramientas a utilizar en la solución del problema planteado.

## **Capítulo 2: Propuesta y características del sistema SAI**

Se realiza una descripción general de la propuesta de aplicación informática, así como sus elementos más importantes. Se especifican los requerimientos funcionales y no funcionales, a partir de los cuales se representan los casos de uso del sistema y la descripción de los mismos.

## **Capítulo 3: Análisis y diseño e implementación del sistema SAI**

Se describe el patrón de arquitectura utilizado en el diseño de la aplicación, se define los elementos y la estructura del análisis y el diseño; y se describe el Modelo de Implementación.

## **Capítulo 1: Fundamentación teórica de la investigación**

En el presente capítulo se realiza un análisis de algunos sistemas existentes en el mundo relacionados con la investigación en curso. Se realiza una caracterización de las tecnologías, herramientas y metodologías propuestas a utilizar en la solución del problema planteado.

### **1.1 Antecedentes**

Después de realizar una investigación de los sistemas existente en el mundo relacionados con el aprendizaje del idioma Inglés, se encontraron algunos que se describen a continuación.

#### **1.1.1 Sistemas existentes en el ámbito internacional**

##### **Games to learn English**

Es un sitio con numerosos juegos para el aprendizaje y la práctica de las habilidades de comprensión auditiva, lectora y expresión escrita del idioma Inglés, realizados en flash por el Profesor Owen Dwyer de la Universidad de Tecnología de Hefei en Anhui China. (Dwyer, 2010)

##### **ABC Spelling**

ABC Spelling es un recurso completamente didáctico, está diseñado para todas aquellas personas que quieren introducirse en el mundo del Inglés sin ningún tipo de conocimiento previo, o bien para los alumnos que presenten problemas en la escritura de este idioma. Este recurso didáctico ofrece todas las letras del alfabeto en Inglés con sus respectivos nombres, de manera que los usuarios puedan conocerlas bien antes de empezar con las actividades. (Segura, 2010)

Una vez que se hayan conocido todas las letras, los usuarios se pueden dedicar a las actividades que propone este programa, entre las que se encuentran: la discriminación de letras, actividades de pronunciación o de deletreo, de reconocimiento y muchas otras. Cuando no se sabe resolver una de las actividades que el programa plantea, se debe acceder a su barra de herramientas, encontrándose la solución al ejercicio. (Segura, 2010)

### **Rosetta Stone**

Rosetta Stone es un conjunto de software didácticos que incluye aplicaciones web y de escritorio, producido por la empresa Rosetta Stone Inc para contribuir al aprendizaje de numerosos idiomas como: Inglés, Alemán, Francés, Italiano, Portugués, Ruso, Polaco, Español, Holandés y muchos más. Este software utiliza una combinación de imágenes, texto y sonido, donde los niveles de dificultad suben a medida que el estudiante avanza. En estas lecciones el estudiante aprende vocabulario y funciones gramaticales sin traducción alguna. El objetivo del programa es que los estudiantes aprendan el idioma que estudian de la manera que aprendieron su idioma materno. En este se desarrollan las habilidades de comprensión auditiva y lectora, expresión oral y escrita. (Rosetta Stone, 2010)

Rosetta Stone permite desarrollar una base de vocabulario fundamental y estructuras esenciales de los idiomas. Las lecciones incluyen temas de la vida cotidiana como: relaciones familiares, edad, decir la hora, términos de calendario, el clima, indicaciones, ubicaciones, cenar en restaurantes, expresar emociones, opiniones e ideas, términos de política, medios de comunicación y comerciales, conversaciones sobre arte y turismo. (Rosetta Stone, 2010)

### **1.1.2 Sistemas existentes en el ámbito nacional**

#### **Rainbow**

Este software corresponde a la asignatura Inglés que se imparte en los tres grados del nivel Secundaria Básica. Se trabaja sobre la base de tres conceptos fundamentales: “vocabulario”, “gramática” y “funciones comunicativas”, que se integran para lograr la comunicación tanto oral como escrita. Estos temas están divididos a su vez en 59 subtemas con 53 epígrafes y 2234 hipervínculos. Presenta 452 ejercicios con igual cantidad de retroalimentaciones reflexivas y 5 juegos: sopa de letras, parchís, crucigrama, karaoke y descubre la imagen. En la biblioteca del software aparecen 1256 palabras con sus significados, con 410 de ellas ilustradas, 301 fotos y 18 videos comentados. Contiene dieciocho temáticas de actualización para los docentes. Brinda servicios informáticos de búsqueda, impresión, copia de textos e imágenes a otras aplicaciones. Por ser un sistema abierto permite la actualización de la base de preguntas y juegos. (Cubaeduca, 2009)

## Sunrise

El software Sunrise forma parte de la Colección Futuro, producida por la Red de Centros de Estudio de Software Educativo del Ministerio de Educación de la República de Cuba (InstEd), con el objetivo de constituir un medio de apoyo al proceso de enseñanza-aprendizaje para el nivel medio superior o bachillerato, del sistema educacional cubano. (InstEd, 2009 )

Este software corresponde a la asignatura Inglés y para su estudio se organizaron los temas a partir del programa de la asignatura, mediante un sistema hipertexto que posee los siguientes capítulos (InstEd, 2009 ) organizados en un menú interactivo:

- a) Funciones comunicativas
- b) Vocabulario
- c) Gramática
- d) Las pistas textuales

Cada tema posee diferentes contenidos, estos tienen una progresividad temática con respecto a los presentados en Rainbow (software destinado a los estudiantes de Secundaria Básica). (InstEd, 2009 )

## Entorno Virtual de Aprendizaje (EVA)

El EVA está configurado sobre la plataforma Moodle, quedando personalizado para la universidad, al cual se le mantuvo el mismo nombre de “Entorno Virtual de Aprendizaje”. El mismo se utiliza para impartir las diferentes asignaturas a los estudiantes, entre ellas el idioma Inglés. El EVA brinda diferentes funcionalidades así como facilidades para impartir clases a distancia y en la categoría semi-presencial o presencial (León, 2010 pág. 1).

Después de realizado el análisis de los sistemas informáticos a nivel internacional y nacional, se presentan a continuación las principales características de estos sistemas.

Tabla 2. Características de los sistemas

|                               |  |
|-------------------------------|--|
| <b>Games to learn english</b> | Practica las habilidades comprensión auditiva, lectora y expresión escrita; pero solamente en juegos en flash. |
| <b>ABC spelling</b>           | Solo practica la habilidad de expresión escrita.   |



|                      |  |
|----------------------|--|
| <b>Rainbow</b>       | Se practica las habilidades de expresión oral y escrita.                                     |
| <b>Sunrise</b>       | Posee diferentes capítulos para el desarrollo de las habilidades de expresión oral y escrita |
| <b>EVA</b>           | Se desarrollan las habilidades de comprensión auditiva y lectora, expresión oral y escrita.  |
| <b>Rosetta Stone</b> | Se desarrollan las habilidades de comprensión auditiva y lectora, expresión oral y escrita.  |

Ninguno de estos sistemas se puede utilizar como propuesta de solución al problema planteado, debido a que no abarcan temas relacionados con la producción (no son de código abierto, imposibilitando su modificación) y no permiten crear ejercicios diferenciados, teniendo en cuenta el nivel de conocimiento de cada estudiante. El EVA podría ser una solución, pero no cumple con unas de las principales características que se quiere lograr con dicha investigación (ser un sistema portable, sin tener que instalar otras herramientas para su uso).

### 1.2 Tecnologías, metodologías y herramientas

A continuación se describen algunas de las tecnologías, y herramientas posibles a utilizar en la solución del problema:

#### 1.2.1 C Sharp v4.0

C Sharp o C# es un lenguaje de programación orientado a objetos, desarrollado por la empresa Microsoft. Este lenguaje combina las mejores características de lenguajes como C, C++ y Java. (Ferguson, y otros, 2003 pág. 37)

**Características de C Sharp** (Ferguson, y otros, 2003 págs. 37-51):

- **Sencillez de C Sharp:** elimina muchos elementos que otros lenguajes incluyen y que son innecesarios en .NET.
- **Modernidad:** C Sharp incorpora en el propio lenguaje, elementos que son muy útiles para el desarrollo de aplicaciones.
- **Orientación a objetos:** C Sharp es un lenguaje orientado a objetos. Soporta características como, encapsulación, herencia y polimorfismo.

- **Orientado a componentes:** la propia sintaxis de C Sharp incluye elementos propios del diseño de componentes, que otros lenguajes tienen que simular e incluye formas de definir propiedades, eventos o atributos.
- **Recolección de basura:** todo lenguaje incluido en la plataforma .NET tiene a su disposición el recolector de basura del CLR. Esto implica, que no es necesario incluir instrucciones de destrucción de objetos en el lenguaje.
- **Seguridad de tipos:** C Sharp incluye mecanismos de control de acceso a tipos de datos, lo que garantiza que no se produzcan errores difíciles de detectar como un acceso a memoria de ningún objeto. (Ferguson, y otros, 2003)

### 1.2.2 JAVA

Java es un lenguaje orientado a objeto de una plataforma independiente. Esta programación tiene mucha similitud con los lenguajes como C y C++. Con la programación en Java, se pueden realizar distintas aplicativos, como son applets (programas pequeños y de propósitos específicos). (Java, 2009)

### 1.2.3 XML v1.0

Lenguaje de Marcado Extensible (XML), es una tecnología muy sencilla que tiene a su alrededor otras tecnologías que la complementan y lo hacen mucho más grande. XML permite compartir los datos con los que se trabaja por todas las aplicaciones y soportes. El uso del XML permite compartir información de forma segura y realizar el intercambio de documentos entre las aplicaciones, tanto en el propio PC como en una red local o extensa. (XML, 2010)

### 1.2.4 UML v2.1

Lenguaje Unificado de Modelado (UML) es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema. Este ofrece un estándar para describir un plano del sistema, incluyendo aspectos conceptuales tales como: procesos de negocio, funcionalidades del sistema, esquemas de base de datos y componentes de software reutilizables. El lenguaje UML tiene una notación gráfica muy expresiva que permite representar en mayor o menor medida todas las fases de un proyecto informático: desde el análisis con los casos de uso, el diseño con los diagramas de clases, objetos, etc., hasta la implementación y configuración con los diagramas de despliegue. (Orallo, 2007 pág. 1)

### 1.2.5 Proceso Unificado de Desarrollo

Proceso Unificado de Desarrollo (RUP, por sus siglas en inglés) es una infraestructura flexible de desarrollo de software, que proporciona prácticas recomendadas probadas y una arquitectura configurable. (GSInnova, 2012)

RUP está basado en componentes e interfaces bien definidas, y junto con UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. RUP se divide en cuatro fases, dentro de las cuales se realizan varias iteraciones según el proyecto: Fase de Inicio, Fase de Elaboración, Fase de Construcción, Fase de Transición. (Figueroa, y otros, 2011)

**Sus características principales** (Figueroa, y otros, 2011) **son:**

- Unifica los mejores elementos de metodologías anteriores
- Preparado para desarrollar, pequeños, grandes y complejos proyectos
- Orientado a objetos
- Utiliza el UML como lenguaje de representación visual

### 1.2.6 Arquitectura en Tres Capas

La arquitectura en tres capas se encarga de encapsular las reglas del negocio asociadas con el sistema y las separa de la presentación y del código de la base de datos, las capas se dividen en (Rios, 2010):

- Capa de presentación: con la interfaz de usuario (formularios windows, páginas HTML) y los controles visuales (textBox, comboBox, dataGrids) junto con los eventos (los click y etc.).
- Capa de negocio (lógica del dominio): aquí se incorpora todo el código que define las reglas de negocio (cálculos, validaciones). Surge de los procesos encontrados en el análisis.
- Capa de acceso a datos: el código que permite acceder a las fuentes de datos. Esencialmente trata sobre cuatro operaciones básicas, llamadas Create-Retrieve-Update y Delete (CRUD), que se realizan sobre cualquier fuente de datos (normalmente alguna base de datos relacional).

### 1.2.7 Modelo Vista Controlador

Modelo Vista Controlador (MVC) es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos. (Arrocha, 2008)

Este se divide en tres partes (Yanette, 2012 pág. 2):

- **Modelo:** representa específicamente la información con la cual el sistema va a trabajar. La lógica de datos puede llegar a asegurar la integridad de ellos y permitirá derivar nuevos datos. La capa del modelo se puede dividir en la capa de acceso a los datos y en la capa de abstracción de la base de datos.
- **Vista:** presenta el modelo con el que va a interactuar el usuario, más conocida como interfaz.
- **Controlador:** el controlador responde más bien a eventos, normalmente son acciones que el usuario invoca, implica cambios en el modelo y también en la vista (interfaz).

### 1.2.8 MonoDevelop v3.0

MonoDevelop es un entorno de desarrollo integrado (IDE) diseñado para programar en C Sharp y otros lenguajes .NET. Las características principales (Guia-ubuntu, 2011) son:

- Completado de código: permite completar el tipo, métodos y campos de una clase mientras se escribe
- Ayuda integrada: incluye la documentación de .NET para facilitar la programación
- Soporte para proyectos: MonoDevelop tiene la capacidad de trabajar con proyectos que ayudarán escribir aplicaciones de consola
- Añadidos: a través de esta característica se puede añadir funcionalidades a MonoDevelop, instalando plugins accesibles a través de repositorios en línea

Otras de las características (Monodevelop, 2010) son:

- Multi-plataforma
- Compatible con Linux, Windows y Mac OS X
- Permite edición avanzada de texto
- Permite código de finalización de soporte para C #, plantillas de código, plegado de código
- Banco de trabajo configurable
- Presenta soporte para múltiples idiomas C #, Visual Basic.Net, C / C + +
- Posibilita crear proyectos web con soporte completo, completado de código y prueba en XSP, el servidor web Mono.

### 1.2.9 Netbeans IDE v6.9

NetBeans IDE es un entorno de desarrollo. Herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el NetBeans IDE. Este, es un producto libre y gratuito sin restricciones de uso. Dispone de soporte para crear interfaces gráficas de forma visual, desarrollo de aplicaciones web, control de versiones, colaboración entre varias personas, creación de aplicaciones compatibles con teléfonos móviles, resaltado de sintaxis. (Ramírez, 2011)

### 1.2.10 Enterprise Architect v7.1

Enterprise Architect (EA), es una herramienta comprensible de diseño y análisis UML, que cubre el desarrollo de software desde el paso de los requerimientos a través de las etapas del análisis, modelo de diseño, pruebas y mantenimiento. EA es una herramienta multi-usuario, basada en Windows, diseñada para ayudar a construir software robusto y fácil de mantener. Ofrece salida de documentación flexible y de alta calidad. Proporciona trazabilidad completa desde el análisis de requerimientos y los artefactos de diseño, hasta la implementación y el despliegue. (Sparxsystems, 2008)

**Entre las principales características con las que cuenta (Sparxsystems, 2008) están:**

- Velocidad, estabilidad y rendimiento
- Trazabilidad de extremo a extremo
- Construido sobre la base de UML 2.1
- Permite crear elementos del modelo UML para un amplio alcance de objetivos
- Permite ubicar esos elementos en diagramas y paquetes
- Posibilita crear conectores entre elementos
- Posibilita documentar los elementos que ha creado

### 1.2.11 SQLiteManager v0.7

SQLiteManager es un sistema gestor de base de datos SQLite, con una interfaz muy clara, dividida en pestañas para los elementos de diseño, administración y la elaboración de instrucciones Structured Query Language (SQL), con la posibilidad de crear y navegar por las tablas, índices y vistas, insertar, eliminar y editar las tablas, ejecutar sentencias SQL. (Equipo de Softonic , 2011 )

Dispone de un sistema completo de generación de informes, exportables en una gran variedad de formatos, incluyendo HTML, CSV y XML. En definitiva, SQLiteManager ofrece una forma más amena para navegar entre los objetos de base de datos, gestionar las bases de datos y construir instrucciones SQL, junto a una tabla completa con los resultados de las peticiones formuladas. (Equipo de Softonic , 2011 )

### 1.2.12 pgAdmin v1.6

pgAdmin es una herramienta para la administración de base de datos PostgreSQL. Está diseñado para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. El interfaz gráfico soporta todas las características de PostgreSQL y facilita enormemente la administración. La aplicación también incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor, un agente para lanzar scripts programados. La conexión al servidor puede hacerse mediante conexión TCP/IP o Unix Domain Sockets (en plataformas nix), y puede cifrarse mediante SSL para mayor seguridad. (Guia-ubuntu, 2008)

### 1.2.13 Oracle Berkeley DB (BDB)

Oracle Berkeley DB (BDB) es un robusto Sistema Gestor de Base de Datos (SGBD) embebidas, desarrollado en C, compatible con los lenguajes Java, Python, PHP, Ruby, TCL C++ y C. Este SGBD soporta tareas multiproceso y múltiples hilos por proceso, transacciones distribuidas, replicación y respaldos en frío y caliente. (Carretero, y otros, 2008)

Otras de las características que presenta (Carretero, y otros, 2008):

- Recuperación de datos secuencial e indexada
- Codificación de datos mediante el algoritmo de cifrado simétrico Advanced Encryption Standard (AES)
- BDB admite sólo las operaciones lógicas insertar, eliminar, buscar y actualizar registros en las tablas; además no es un SGBD relacional y no admite consultas SQL.

### 1.2.14 MongoDB

MongoDB es un SGBD No SQL orientado a documentos y desarrollado en C++. Es compatible con C, C++, C#, Erlang, Haskell, Java, JavaScript, Lisp, node.JS, Perl, PHP, Python, Ruby y Scala. Soporta la búsqueda por campos, consultas de rangos y expresiones regulares. Cualquier campo en un documento de MongoDB puede ser indexado, al igual que es posible hacer índices secundarios. MongoDB admite

también procesamiento por lotes de datos y operaciones de agregación; tiene la capacidad de realizar consultas utilizando JavaScript, haciendo que estas sean enviadas directamente a la base de datos para ser ejecutadas. (MongoDB, 2010)

## 1.2.15 SQLite v3.7

SQLite es una librería compacta, autocontenida de código abierto y distribuida bajo dominio público que implementa un SGBD relacional embebido. Este almacena la base de datos en un único fichero, por lo que no necesita ningún tipo de configuración y admite consistencia de base de datos, aislamiento, durabilidad, triggers. (Montiel, 2008)

**Ventajas de SQLite** (Hipp, 2011):

- **Tamaño:** tiene una pequeña memoria y una única biblioteca necesaria para acceder a base de datos, lo que lo hace ideal para aplicaciones de base de datos incorporadas.
- **Rendimiento de base de datos:** realiza operaciones de manera eficiente y es más rápido que MySQL y PostgreSQL.
- **Portabilidad:** se ejecuta en muchas plataformas y sus bases de datos pueden ser fácilmente portadas sin ninguna configuración o administración.
- **Interfaces:** cuenta con diferentes Interfaz de Programación de Aplicaciones (API), las cuales permiten trabajar con C++, PHP, Perl, Python, Tcl y groovy.
- **Costo:** es de dominio público, y por tanto, es libre de utilizar para cualquier propósito sin costo y se puede redistribuir libremente.

## 1.2.16 PostgreSQL v9.2

PostgreSQL es un sistema de gestión de base de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de base de datos de código abierto más potente del mercado. PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos, para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando. (Postgresql, 2010)(sic)

**Características más generales de PostgreSQL** (Postgresql, 2010):

- Integridad referencial
- Replicación asíncrona / *Streaming replication - Hot Standby*

- Unicode
- Multi-Version Concurrency Control (MVCC)
- Múltiples métodos de autenticación
- Acceso cifrado vía SSL
- Completa documentación
- Disponible para Linux y UNIX en todas sus variantes (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows 32/64bit

### 1.3 Herramientas, metodología y tecnologías propuestas para la solución del problema planteado

Después de haber realizado un estudio del arte de las diferentes tecnologías, herramientas y lenguajes, se propone utilizar:

Como lenguaje de programación C Sharp v4.0 por:

- Su sencillez
- Modernidad
- Orientación a objetos
- Orientación a componentes
- Instrucciones seguras
- Se presenta conocimiento del mismo

Como lenguaje de marcas extensibles XML v1.0 porque:

- La principal novedad consiste en permitir compartir los datos con los que se trabaja, por todas las aplicaciones
- Permite guardar los datos de configuración del usuario, así como sus trabajos realizados
- Comparte la información de una manera segura, fiable y fácil
- Permite integrar datos de las fuentes más dispares

Como metodología de desarrollo RUP porque:

- Es la que se conocía, además de que se necesitaba un sistema en tiempo rápido y aprender otra metodología iba a llevar mucho más tiempo



- Cumple con las necesidades para realizar el sistema
- Es la que se enseña en la UCI
- Está destinado tanto para el desarrollo de proyectos grandes como para los pequeños
- Es centrado en la arquitectura
- Guiado por casos de uso
- Iterativo e incremental y en cada ciclo de iteración, se hace exigente el uso de artefactos
- Está definida en la universidad como metodología de desarrollo de software
- Es la más utilizada a nivel mundial por su estructura interna

Como herramienta de modelado Enterprise Architect v7.1 porque:

- Presenta facilidad de uso, velocidad, estabilidad y rendimiento
- Construido sobre la base de UML 2.1
- Soporta muy bien la metodología seleccionada
- Es una de las más usadas en la UCI
- Se presenta un buen conocimiento de la misma

Como lenguaje de modelado UML v2.1 porque:

- Es orientado a objetos
- Tiene una notación gráfica muy expresiva, que permite representar en mayor o menor medida todas las fases de un proyecto informático
- Es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software
- Permite especificar cuáles son las características del sistema antes de su construcción
- Es el lenguaje que soporta la herramienta CASE seleccionada

Como herramienta de desarrollo MonoDevelop v3.0 porque:

- Es multi-plataforma diseñado principalmente para C # y otros lenguajes .NET
- Compatible con Linux, Windows y Mac OS X
- Presenta edición avanzada de texto

- Presenta código de finalización de soporte para C #, plantillas de código y plegado de código

Como patrón arquitectónico Modelo Vista Controlador porque:

- Es posible tener diferentes vistas para un mismo modelo
- Es posible construir nuevas vistas sin necesidad de modificar el modelo subyacente
- Proporciona un mecanismo de configuración a componentes complejos mucho más tratables que el puramente basado en eventos (el modelo puede verse como una representación estructurada del estado de la interacción)

Como sistemas gestores de base de datos:

SQLite v3.7 porque:

- De los analizados es el único SGBD relacional. Esto permite realizar relaciones entre las tablas y favorece la normalización de las bases de datos, garantizando la integridad de los datos porque no existirá información redundante ni omitida.
- Las tablas al estar normalizadas permiten consultas rápidas y eficaces.
- Existirá menor número de índices por tabla, de esta forma se mejora el rendimiento de las instrucciones insert, update y delete.
- Los índices serán más estrechos y compactos, dando así, mayor rapidez en la ordenación y creación de estos.
- A diferencia de los otros SGBD embebidas, en SQLite se realizan las consultas con SQL, esta característica ofrece una gran ventaja, puesto que las consultas realizadas pueden ser reutilizadas para acceder a bases de datos de PostgreSQL u otros SGBD que soporten consultas en SQL.
- La aplicación se quiere que sea portable, y este sistema permite la gestión de base de datos portables.

PostgreSQL v9.2 porque:

- Está definido por el Departamento SAS y se tiene experiencia en el trabajo con este SGBD. Se ha utilizado en numerosas ocasiones para el desarrollo de otras bases de datos y esto evita que se emplee más tiempo analizando y aprendiendo otros SGBD, impidiendo que los resultados de la investigación se obtengan a tiempo.

- Utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema, por lo que si ocurre un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.
- La capacidad de información que permiten almacenar las bases de datos es ilimitado, solo depende del sistema de almacenamiento que se utilice.
- Es capaz de manejar complejas rutinas y reglas, permite el control de concurrencia multi-versión, soporte multi-usuario, transacciones, optimización de consultas y herencia.

Como herramientas de administración de base de datos SQLite Manager 0.7 porque:

- Cuenta con una interfaz muy clara, dividida en pestañas para los elementos de diseño, administración y la elaboración de instrucciones SQL.
- Tiene la posibilidad de crear y navegar por las tablas, índices y vistas, insertar, eliminar y editar las tablas, así como ejecutar sentencias SQL.
- Dispone de un sistema completo de generación de informes, exportables en una gran variedad de formatos, incluyendo HTML, CSV y XML.

pgAdmin III porque:

- Está diseñado específicamente para administrar base de datos en PostgreSQL.
- Está diseñado para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas.

## Capítulo 2: Propuesta y características del sistema SAI

En el presente capítulo se realiza una descripción de la aplicación que se propone desarrollar, así como sus elementos más importantes. Se desarrolla el modelo de dominio, se especifican los requisitos funcionales y no funcionales, a partir de los cuales se representan los casos de uso del sistema y la descripción de los mismos; así como los actores que interactúan con la aplicación.

### 2.1 Descripción de la aplicación propuesta

Sobre la base del análisis realizado a los problemas existentes en los proyectos productivos del Departamento SAS, relacionados con el aprendizaje del idioma Inglés; se propone desarrollar una aplicación informática. La misma permitirá contribuir al desarrollo de la habilidad de comprensión lectora en el idioma Inglés, de los estudiantes de dicho departamento, con posibilidades de extenderse a otros departamentos o centros de la UCI.

Después de realizada una encuesta a los estudiantes del Departamento SAS se determinó que estudiarán, al contar con una aplicación de escritorio que presente información en Inglés de temas productivos. Se obtuvo que: el 69% de los encuestados si estudiarían, un 6% no lo haría y un 25% no sabe; como se muestra en la figura 1.

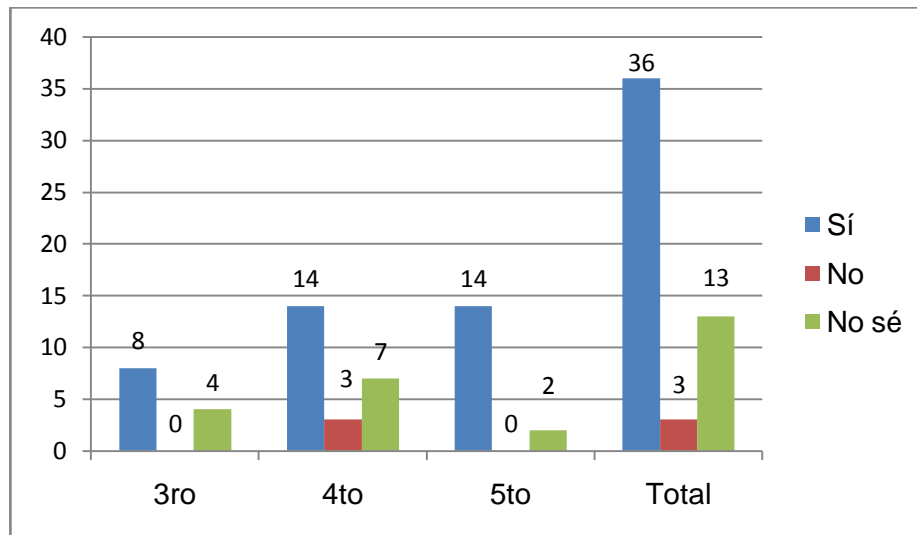


Figura 1. Cantidad de estudiantes que estudiarían con una aplicación de escritorio (elaboración propia)

Los temas que los estudiantes prefieren estudiar en la producción son: las tecnologías, lenguajes de programación, herramientas y negocios en un menor porcentaje. Como se puede apreciar en la figura siguiente:

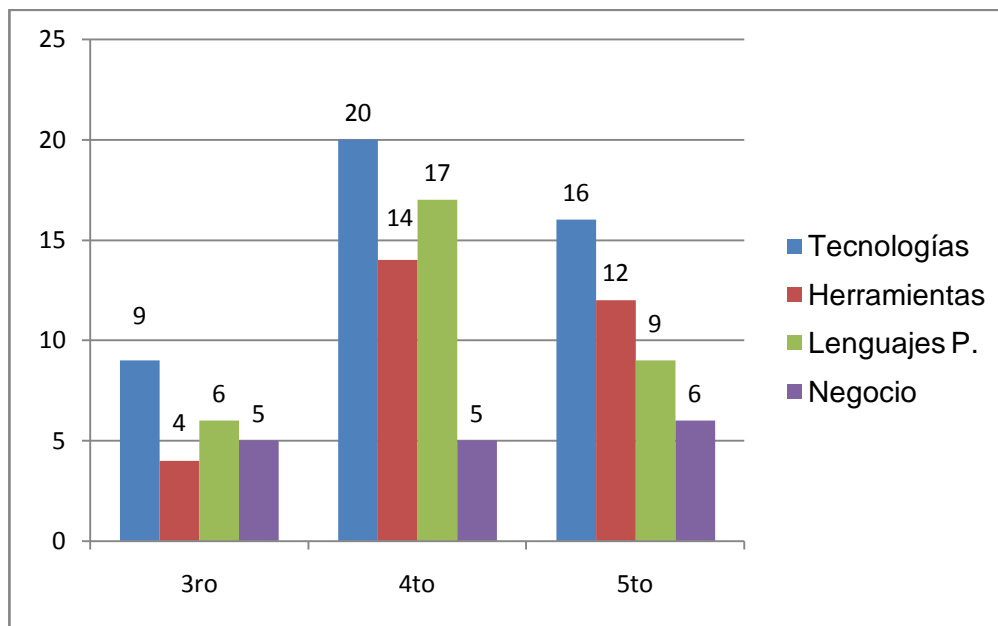


Figura 2. Temas preferidos por los estudiantes (elaboración propia)

La mayoría de los estudiantes desean realizar todos los tipos de actividades, como se puede observar en la figura siguiente:

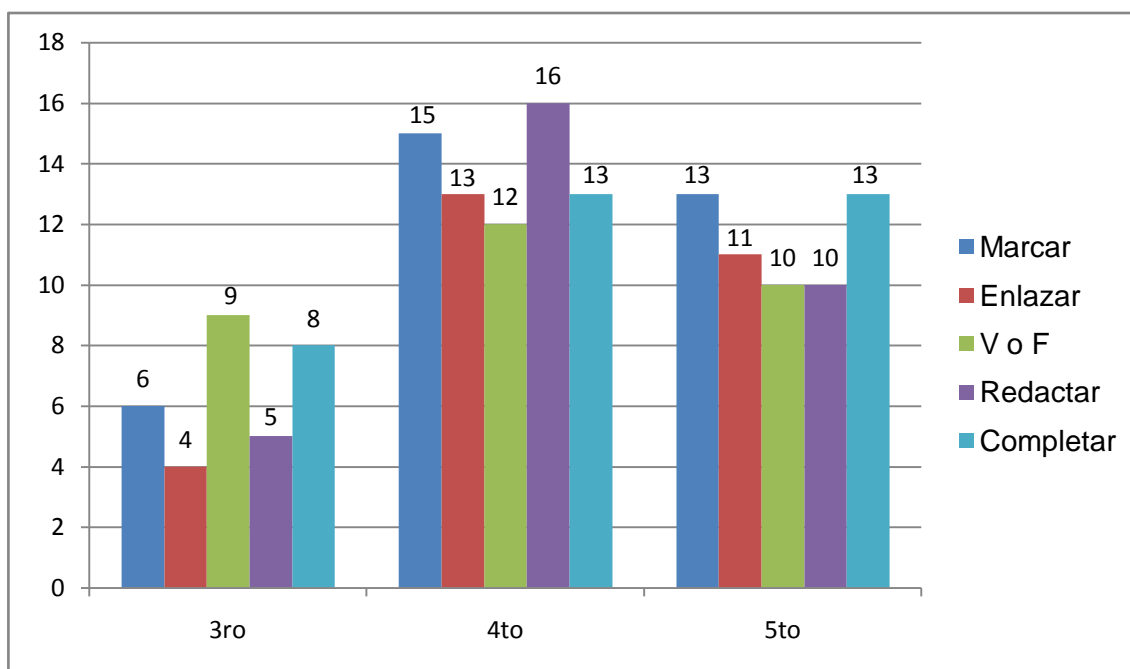


Figura 3. Tipo de actividades que los estudiantes prefieren (elaboración propia)

La encuesta también demostró que la mayoría de los estudiantes prefieren una aplicación de escritorio, porque: se podría utilizar incluso cuando no haya conexión, desde cualquier lugar de la universidad, entre otras opiniones.

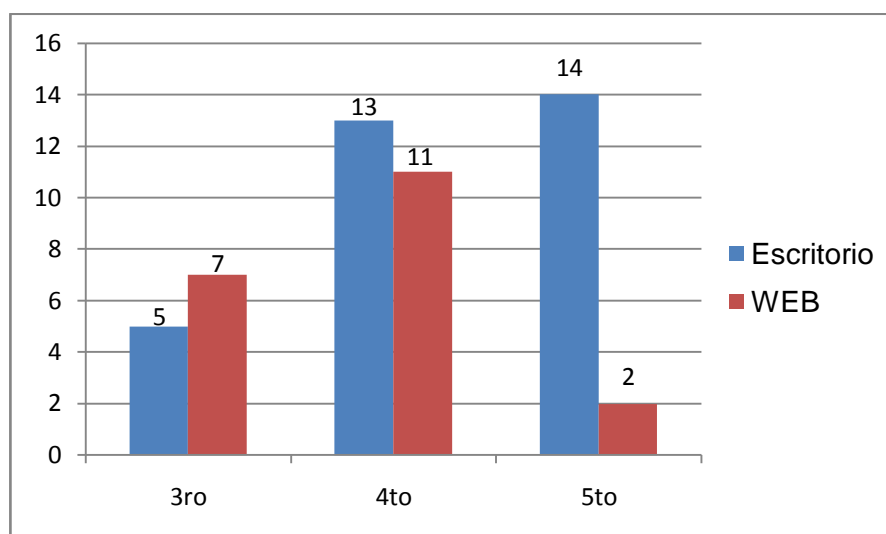


Figura 4. Tipo de aplicación que prefieren los estudiantes (elaboración propia)

Basándose en los resultados que arrojó la encuesta, se propone la aplicación informática llamada Sistema de Aprendizaje para el idioma Inglés (SAll), esta debe contener un conjunto de temas, compuestos por actividades de tipo: marcar, enlazar columnas, verdadero o falso, redactar y completar; cuyos contenidos estarán relacionados con los lenguajes de programación, herramientas y tecnologías que se utilizan en el Departamento SAS.

Con la aplicación podrá interactuar el usuario **Profesor**, encargado de crear los diferentes temas con sus actividades y asignar estas a usuarios **Estudiante** para que sean resueltas. Una vez que los estudiantes resuelvan las actividades de los temas que les fueron asignados, el profesor podrá evaluar la actividad de redacción y otorgarle una nota al tema en general. La aplicación debe permitir su uso desde cualquier lugar de la universidad e incluso fuera de esta. La misma permitirá trabajar desde los laboratorios de producción conectada a una base de datos (PostgreSQL) donde estarán los temas creados y asignados por el profesor. Si los estudiantes desean realizar los temas, que le fueron asignados, en otra computadora que no se encuentra en los laboratorios del Departamento SAS, podrán descargar los mismos en un fichero XML y llevárselos para resolverlos en otra estación de trabajo; para esto la aplicación deberá estar instalada en la misma.

Cuando un estudiante haya resuelto los temas, podrá exportar las respuestas en fichero XML. Este fichero, podrá ser enviado vía correo al profesor o ser cargado en la base de datos (PostgreSQL), que se encuentra en uno de los laboratorios del Departamento SAS, para que el profesor pueda evaluar. De esta manera, la aplicación propuesta permitirá a los estudiantes, incrementar el estudio del idioma Inglés, relacionado con temas de la producción, desde cualquier lugar que necesite.

### **2.2 Modelo del dominio**

Un Modelo de dominio es una representación visual de clases conceptuales o de objetos reales en un dominio de interés. Es la representación de los conceptos más importantes y significativos en el desarrollo de un sistema. Representa clases conceptuales, conceptos del mundo real y no de los componentes del software. Su principal objetivo es ayudar a comprender los conceptos que utilizan los usuarios, las definiciones con las que trabajan y con las que deberá trabajar la aplicación. (Martínez, 2009)

### 2.2.1 Conceptos fundamentales

Para el entendimiento del diagrama del modelo de dominio que se muestra a continuación, se proporciona un marco conceptual con algunas definiciones identificadas en la elaboración de dicho modelo.

**Usuario:** representa una persona que interactúa con la aplicación y que presenta cierto grado de acceso a los recursos que presenta la misma. Estos pueden ser estudiantes o profesores.

**Estudiante:** es el usuario que más va a interactuar con la aplicación realizando los ejercicios de los temas asignados.

**Profesor:** es el encargado de crear los temas y actividades a desarrollar por el estudiante. Además tiene la responsabilidad de revisar y evaluar las actividades realizadas por el estudiante.

**Autenticar:** tipo de acción que le permite a un usuario acceder a la aplicación.

**Aplicación:** representa un tipo de programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de trabajos.

**Temas:** representa la materia o idea a tratar en las actividades a realizar en la aplicación. (rae, 2005)

**Actividades:** tareas que el usuario puede realizar en la aplicación relacionada con los temas asignados por el profesor.

**Incisos:** representa la cantidad de ejercicios que tiene una actividad.

**Respuesta:** representa la solución que el usuario le otorga al conjunto de incisos que una actividad presenta.

**Evaluación:** cantidad de puntos que se le otorga a una respuesta. Estas pueden ser cuantitativas o cualitativas.

**Cuantitativa:** tipo de evaluación que se le da a una respuesta numéricamente.

**Cualitativa:** tipo de evaluación que se le da a una respuesta, con letras.





### 2.3.1 Requisitos Funcionales (RF)

Los requerimientos funcionales de un sistema describen lo que el sistema debe hacer. Estos requerimientos dependen del tipo de software que se desarrolle, de los posibles usuarios del software y del enfoque general tomado por la organización al redactar requerimientos. (Sommerville, 2005)

A continuación se muestra un listado con los RF definidos para la aplicación así como la descripción de cada uno:

Tabla 3. Requisitos Funcionales (elaboración propia)

| Requisitos Funcionales              |                                       |
|-------------------------------------|---------------------------------------|
| RF1 Autenticar                      | RF2 Crear Perfil                      |
| RF3 Modificar Perfil                | RF4 Crear Tema                        |
| RF5 Listar Tema                     | RF6 Modificar Tema                    |
| RF7 Buscar Tema                     | RF8 Eliminar Tema                     |
| RF9 Crear Actividad                 | RF10 Buscar Actividad                 |
| RF11 Listar Actividad               | RF12 Modificar Actividad              |
| RF13 Eliminar Actividad             | RF14 Exportar XML por Tema estudiante |
| RF15 Buscar Evaluación              | RF16 Importar XML por Tema estudiante |
| RF17 Exportar XML por Tema profesor | RF18 Importar XML por Tema profesor   |
| RF19 Buscar Respuesta               | RF20 Realizar Actividad               |
| RF21 Insertar Respuesta             | RF22 Evaluar                          |
| RF23 Buscar Estudiante              | RF24 Insertar Evaluaciones            |
| RF25 Listar Evaluación              | RF26 Asignar Tema                     |

#### Descripción de los RF:

**RF1 Autenticar:** permite verificar las credenciales del usuario para acceder a la aplicación.

**RF2 Crear Perfil:** permite al usuario crear un nuevo perfil.

**RF3 Modificar perfil:** permite al usuario realizar modificaciones en la información de un perfil.

**RF4 Crear Tema:** permite al usuario crear nuevos temas en la aplicación.

**RF5 Listar Tema:** la aplicación muestra un listado de los temas creados.

**RF6 Modificar Tema:** permite al usuario realizar modificaciones en la información de los temas.

- RF7 Buscar Tema:** permite al usuario visualizar los temas con que cuenta la aplicación.
- RF8 Eliminar Tema:** permite al usuario eliminar un tema.
- RF9 Crear Actividad:** permite al usuario crear nuevas actividades en la aplicación.
- RF10 Buscar Actividad:** permite al usuario visualizar las actividades con que cuenta la aplicación.
- RF11 Listar Actividad:** la aplicación muestra un listado de las actividades.
- RF12 Modificar Actividad:** permite al usuario realizar modificaciones en la información de las actividades.
- RF13 Eliminar Actividad:** permite al usuario eliminar una actividad.
- RF14 Exportar XML por Tema estudiante:** permite al usuario exportar un XML con los temas realizados.
- RF15 Buscar Evaluación:** permite realizar la búsqueda de las evaluaciones.
- RF16 Importar XML por Tema estudiante:** permite al usuario importar un XML con nuevos temas a realizar.
- RF17 Exportar XML por Tema profesor:** permite al usuario exportar un XML con los temas nuevos a realizar.
- RF18 Importar XML por Tema profesor:** permite al usuario importar un XML con los temas realizados por el estudiante.
- RF19 Buscar Respuesta:** permite realizar la búsqueda de las respuestas de las actividades realizadas.
- RF20 Realizar Actividad:** permite al usuario realizar las actividades existentes en la aplicación.
- RF21 Insertar Respuesta:** la aplicación le permite insertar las respuestas de las actividades.
- RF22 Evaluar:** permite evaluar las actividades realizadas.
- RF23 Buscar Estudiante:** permite realizar la búsqueda de un estudiante en la aplicación.
- RF24 Insertar Evaluaciones:** permite insertar las evaluaciones de cada respuesta.
- RF25 Listar Evaluación:** la aplicación muestra un listado de las evaluaciones existentes.
- RF26 Asignar Tema:** la aplicación permite que el profesor asigne los temas a los estudiantes según el nivel de cada uno.

### 2.3.2 Requisitos No Funcionales (RNF)

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Son restricciones de los servicios o funciones ofrecidas por el sistema. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido y confiable. Son fundamentales en el éxito del producto y normalmente están vinculados a requerimientos funcionales. (Sommerville, 2005)

### ➤ Usabilidad

**RNF1** La aplicación podrá ser utilizada por todos los usuarios que tengan autorización.

**RNF2** El usuario Profesor debe tener conocimientos básicos de informática.

**RNF3** La aplicación debe presentar un acceso fácil y rápido, para facilitar el uso de la misma.

**RNF4** El usuario final deberá recibir una capacitación mínima de un día.

### ➤ Disponibilidad

**RNF5** La aplicación estará disponible a tiempo completo, permitiendo el trabajo de los usuarios y la revisión de las actividades en el momento que se necesite.

### ➤ Eficiencia

**RNF6** La aplicación deberá ser rápida ante las solicitudes de los usuarios.

### ➤ Confidencialidad

**RNF7** La información estará protegida contra accesos no autorizados utilizando mecanismos de autenticación propios.

**RNF8** La autenticación será la primera acción del usuario en la aplicación y consistirá en suministrar nombre de usuario único y una contraseña, que debe ser de conocimiento exclusivo de la persona que se autentica. Si el usuario autenticado no se encuentra registrado se debe reportar un error de acceso.

### ➤ Soporte

**RNF9** Una vez puesta en marcha la aplicación y siendo usada por los usuarios finales, se recogerá toda la información referente a los defectos y/o no conformidades existentes, para incorporar las mejoras sugeridas a la aplicación.

**RNF10** El personal que desarrolla (o encargado de dar soporte) debe contar con el nivel técnico requerido mediante adiestramiento de servicio.

### ➤ Integridad

**RNF11** La información podrá ser modificada solo por personal autorizado.

### ➤ Interfaz de usuario

**RNF12** La aplicación será diseñada con una interfaz sencilla y de fácil entendimiento.

### ➤ Software

**PC cliente y PC servidor de base de datos:**

**RNF13** Sistema operativo Windows XP Service pack 3 o superior.

### ➤ Hardware

**PC cliente:**

**RNF14** Microprocesador a 1.0GHz o superior, 256Mb de memoria RAM o superior, 10Gb disponible en disco duro.

**RNF15** Requiere tarjeta de red (para cuando se conecte a la BD PostgreSQL)

**RNF16** Ordenador Pentium IV

**PC servidor de base de datos:**

**RNF17** 3Gb de memoria RAM o superior, 40Gb disponible en disco duro.

**RNF18** Se requiere tarjeta de red

### ➤ Documentación de usuarios en línea y ayuda del sistema

**RNF19** La aplicación contendrá una ayuda, en la que se explicarán de forma clara los detalles del uso de las distintas funcionalidades, garantizando así, el buen desempeño de los usuarios a la hora de interactuar con la misma.

### ➤ Portabilidad

**RNF20** La aplicación será portable, permitiendo el traslado de la misma de una PC a otra o a través de la red informática y dispositivos de almacenamiento.

### ➤ Seguridad

**RNF21** La aplicación contará con una política de seguridad diseñada para mostrar información en dependencia del usuario que interactúe con ella.

### 2.4 Modelo de casos de uso del sistema

El modelo de casos de uso describe la funcionalidad propuesta del nuevo sistema. Un caso de uso representa una unidad discreta de interacción entre un usuario (humano o máquina) y el sistema. (Sparxsystems, 2007)

#### 2.4.1 Definición de los actores

Un actor es un usuario del sistema. Incluye usuarios humanos y otros sistemas computarizados. Un actor usa un caso de uso para desempeñar alguna porción de trabajo que es de valor para el negocio. El conjunto de casos de uso al que un actor tiene acceso define su rol global en el sistema y el alcance de su acción (Sparxsystems, 2007). A continuación se definen los actores del sistema.

Tabla 4. Definición de los actores (elaboración propia)

| Actor      | Descripción  |
|------------|--|
| Usuario    | Usuario de la aplicación encargado de gestionar la información.  |
| Estudiante | Usuario que interactúa con la aplicación para realizar las funcionalidades existentes.   |
| Profesor   | Usuario que interactúa con la aplicación para evaluar las actividades realizadas por los estudiantes y posee privilegios administrativos para el uso y configuración de la aplicación. |

#### 2.4.2 Diagrama de casos de uso del sistema

Los diagramas de casos de uso, documentan el comportamiento de un sistema desde el punto de vista del usuario. Por lo tanto, los casos de uso determinan los requisitos funcionales del sistema, representando las funciones que un sistema puede ejecutar. (Tello, 2008)

A continuación se muestra el diagrama de casos de uso del sistema, donde se pueden apreciar las relaciones que se establecen entre actores y casos de uso.

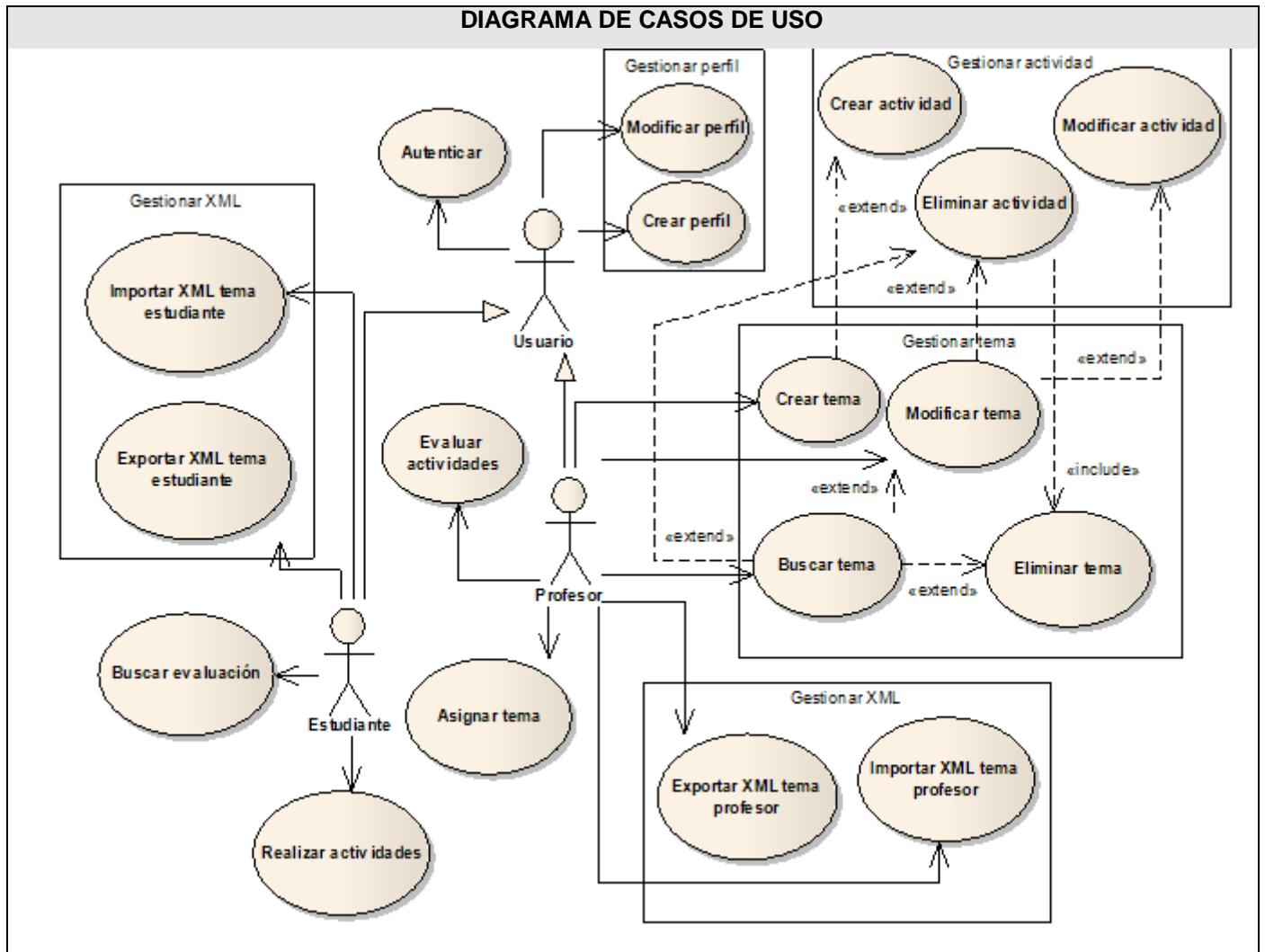


Figura 6. Diagrama de casos de uso del sistema (elaboración propia)

### 2.4.3 Descripción textual de los caso de uso del sistema

A continuación se refleja la descripción textual de los casos de uso arquitectónicamente significativo y la relación de los mismos con los actores.

### CUS. Crear Tema

Tabla 5. Descripción del CUS Crear Tema (elaboración propia)

|                        |  |
|------------------------|--|
| <b>Caso de Uso:</b>    | Crear Tema   |
| <b>Actores:</b>        | <b>Profesor</b>  |
| <b>Resumen:</b>        | El caso de uso inicia cuando el actor accede a la opción Crear Tema, la aplicación brinda la posibilidad de introducir y/o seleccionar los datos para crear el tema, el actor introduce los datos del tema, el sistema crea el tema, el caso de uso termina. |
| <b>Precondiciones:</b> | Debe buscar primero si existe el tema.   |
| <b>Referencias</b>     | RF4, RF5   |

### CUS. Evaluar

Tabla 6. Descripción del CUS Evaluar (elaboración propia)

|                        |   |
|------------------------|---|
| <b>Caso de Uso:</b>    | Evaluar   |
| <b>Actores:</b>        | <b>Profesor</b>   |
| <b>Resumen:</b>        | El caso de uso inicia cuando el actor importa el XML por evaluaciones y accede a la opción Evaluar, el sistema muestra los temas existentes da una evaluación general por tema, el caso de uso termina. |
| <b>Precondiciones:</b> | Debe importar el XML.   |
| <b>Referencias</b>     | RF10, RF22, RF23, RF24, RF25,   |

### CUS. Realizar Actividad

Tabla 7. Descripción del CUS Realizar Actividad (elaboración propia)

|                        |   |
|------------------------|---|
| <b>Caso de Uso:</b>    | Realizar Actividad  |
| <b>Actores:</b>        | <b>Estudiante</b>   |
| <b>Resumen:</b>        | El caso de uso inicia cuando el actor accede a la opción Realizar Actividad, el sistema brinda la posibilidad de seleccionar el tema del que quiera hacer las actividades, el actor selecciona el tema y la actividad a realizar y selecciona la opción resolver actividad, el caso de uso termina. |
| <b>Precondiciones:</b> | El usuario debe estar autenticado.  |
| <b>Referencias</b>     | RF5, RF7, RF10, RF11, RF20  |



### **Capítulo 3: Análisis, diseño e implementación del sistema SAI**

En el presente capítulo se realiza una descripción de la arquitectura del sistema y los patrones arquitectónicos utilizados. Se describen las clases como componentes y subsistemas desarrollados durante el diseño. Se realizan los diagramas de clases del análisis, del diseño y los de interacción, específicamente los de comunicación (de los CUS descritos en el capítulo 2) y el modelo de datos. Se realiza la representación del modelo de implementación, mediante la realización del diagrama de componentes y de despliegue.

#### **3.1 Descripción de la arquitectura**

La arquitectura definida para la solución propuesta, se basa en el patrón arquitectónico Modelo-Vista-Controlador. Este patrón separa el modelado del dominio, la presentación y las acciones basadas en datos ingresados por el usuario en tres componentes diferentes:

- El Modelo estará representado por una base de datos interna y otra remota. En las mismas se manejará y controlará la información con la que trabaja la aplicación.
- La Vista estará compuesta por un conjunto de interfaces visuales que permiten a los usuarios interactuar con la aplicación.
- El Controlador estará definido por la clase Controladora responsable de responder las peticiones que realizan los usuarios al modelo a través de la vista.

#### **3.2 Patrones de Diseño**

El uso de patrones ayuda a obtener un software de calidad. Los patrones de diseño expresan esquemas para definir estructuras de diseño (o sus relaciones) con las que construir sistemas de software. El principal objetivo de estos es encapsular el cambio. (Craig, 1999)

##### **Patrones GRASP**

Los patrones generales para asignar responsabilidades del inglés General Responsibility Assignment Software Patterns (GRASP). Describen los principios fundamentales de diseño de objetos para la asignación de responsabilidades. Constituyen un apoyo para la enseñanza que ayuda a entender el diseño de objeto esencial y aplica el razonamiento para el diseño de una forma sistemática, racional y

explicable. Abarca cinco patrones principales, algunos de estos serán utilizados para contribuir a un mejor desarrollo del módulo (Craig, 1999):

**Patrón Experto:** la clase Controladora, contendrá todos los métodos necesarios para manejar toda la información de la aplicación y las peticiones de los usuarios.

**Patrón Creador:** la clase Controladora será la responsable de crear los objetos de las clases Conexión, Tema, TemaResuelto, Actividad, Inciso, IncisoVoF, IncisoMarcar, IncisoCompletar, IncisoEnlazar e IncisoRedactar.

**Patrón Bajo Acoplamiento:** debido a que la clase Controladora será la responsable de crear los objetos de las clases Conexión, Tema, TemaResuelto, Actividad, Inciso, IncisoVoF, IncisoMarcar, IncisoCompletar, IncisoEnlazar e IncisoRedactar; se garantizará el bajo acoplamiento; si se realizan cambios en una de estas clases, solo habrá que realizar cambios en la clase Controladora.

### Patrones GoF

Los patrones Pandilla de los Cuatro (GoF, por sus siglas en inglés). Proporcionan elementos reusables en el diseño de sistemas de software lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias. (Craig, 1999)

**Patrón Singleton:** es utilizado en la aplicación para garantizar que solo haya una instancia a la vez de los formularios.

## 3.3 Análisis

Para el desarrollo de la aplicación se utiliza la metodología RUP. Dentro de los flujos de trabajo de la misma se encuentra el de Análisis y Diseño. En este flujo se refinan y estructuran los requisitos obtenidos con anterioridad, profundizando en el dominio de la aplicación, lo que permitirá una mayor comprensión del problema para modelar la solución.

### 3.3.1 Modelo de análisis

El modelo de análisis, es la primera representación técnica de un sistema. Este debe cumplir tres objetivos primarios: describir lo que requiere el cliente, establecer una base para la creación de un diseño de

software y definir un conjunto de requisitos que puedan validarse una vez construido el software. (Pressman, 2005)

### 3.1.1 Diagramas de clases del análisis

El diagrama de clases del análisis se caracteriza por centrarse en el tratamiento de los requisitos funcionales. Las clases del análisis se clasifican en:

- ✓ **Clase Interfaz:** modela la interacción entre el sistema y sus actores
- ✓ **Clase Entidad:** modela información persistente
- ✓ **Clase Control:** representa coordinación, secuencia, transacciones y control de otros objetos

A continuación se muestran los diagramas de clases del análisis de los casos de uso arquitectónicamente significativos.

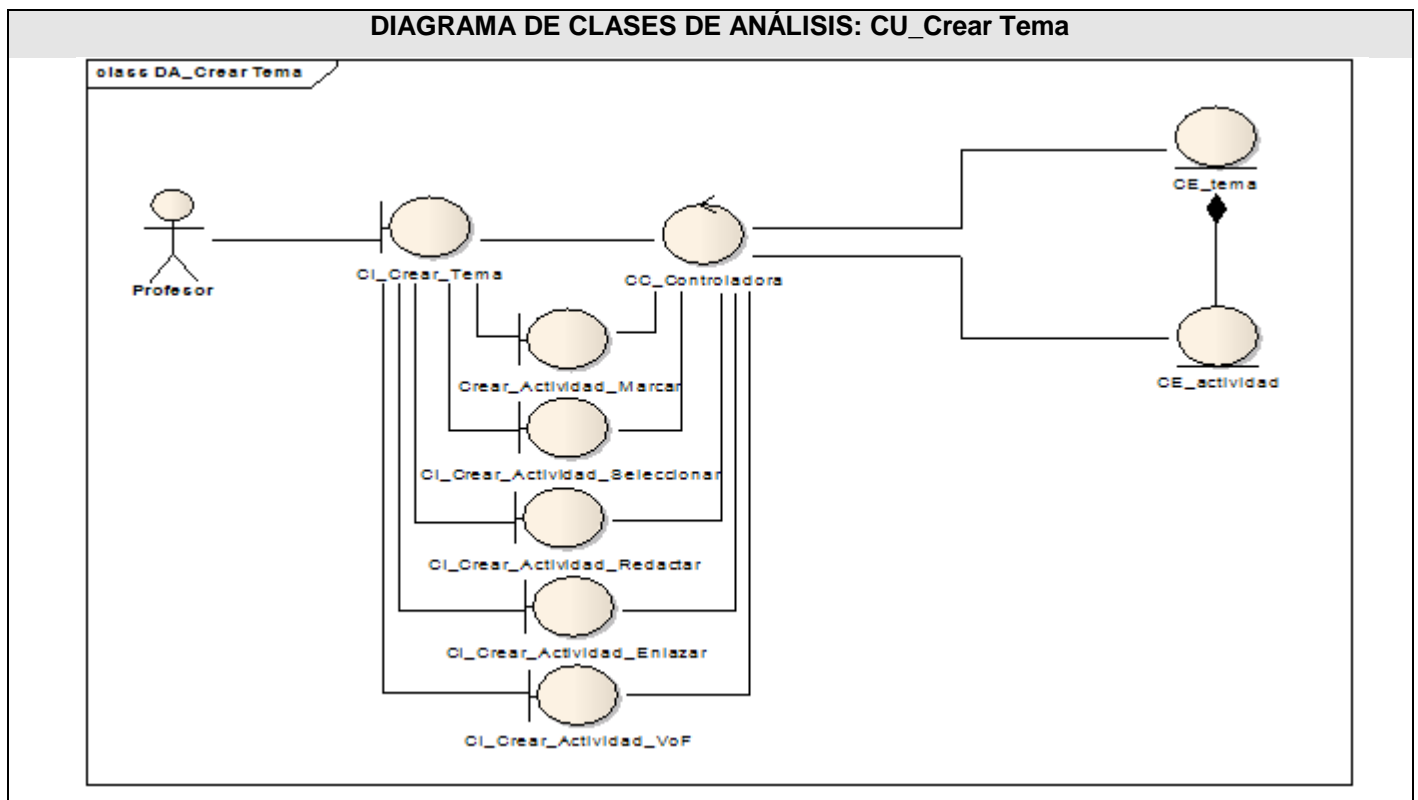


Figura 7. Diagrama de clases de análisis Crear Tema (elaboración propia)

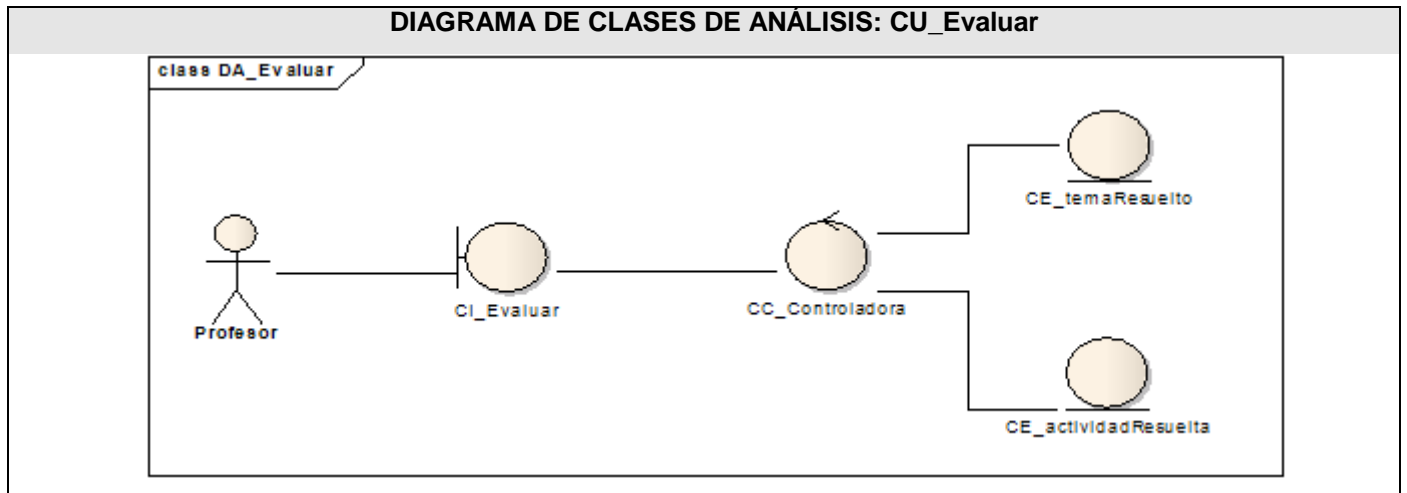


Figura 8. Diagrama de clases de análisis Evaluar (elaboración propia)

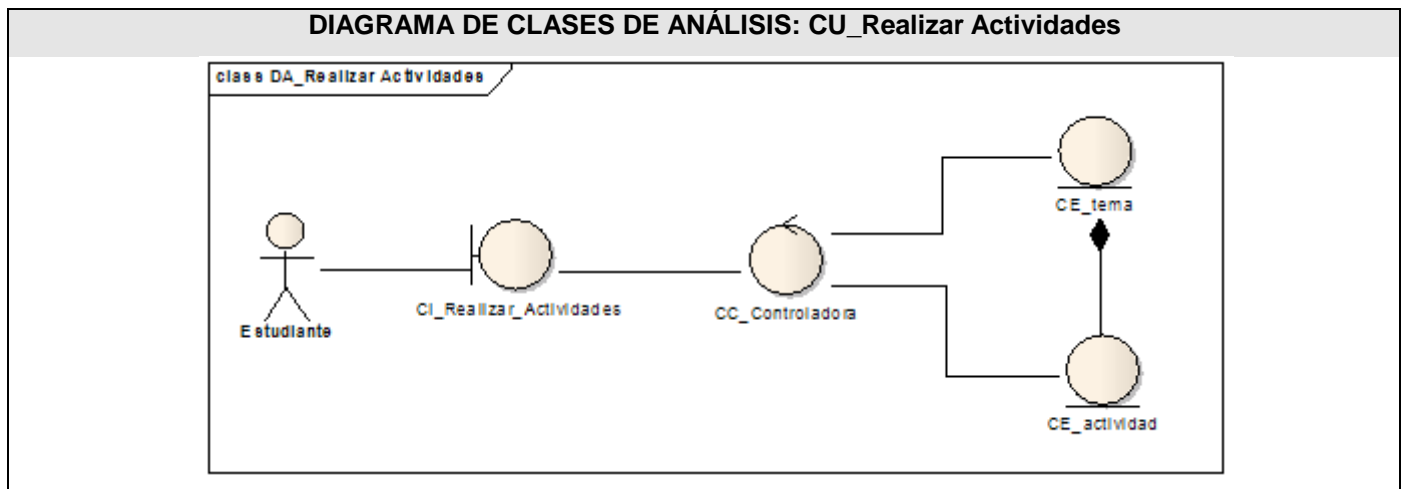


Figura 9. Diagrama de clases de análisis Realizar Actividad (elaboración propia)

### 3.1.2 Diagramas de interacción

Los diagramas de interacción son diagramas que describen, cómo grupos de objetos colaboran para conseguir algún fin. Estos diagramas muestran objetos, así como los mensajes que se pasan entre ellos dentro del caso de uso. Muestran una interacción, que consiste en un conjunto de objetos y sus relaciones, incluyendo los mensajes que puedan ser realizados entre ellos. Son importantes para modelar los aspectos dinámicos de un sistema y para construir sistemas ejecutables a través de ingeniería hacia

adelante e ingeniería inversa. Comúnmente contienen: objetos, enlaces y mensajes. Pueden servir para visualizar, especificar, construir y documentar los aspectos dinámicos de una sociedad particular de objetos, o pueden ser usados para modelar un flujo particular de control de un caso de uso. Los diagramas de interacción capturan el comportamiento de los casos de uso. Estos están conformados por los diagramas de secuencia y los de comunicación (UML, 2006):

**Diagramas de comunicación:** muestran las relaciones entre los objetos y los mensajes que intercambian.

**Diagramas de secuencia:** muestran las interacciones expresadas en función de secuencias temporales.

### 3.1.2.1 Diagramas de comunicación

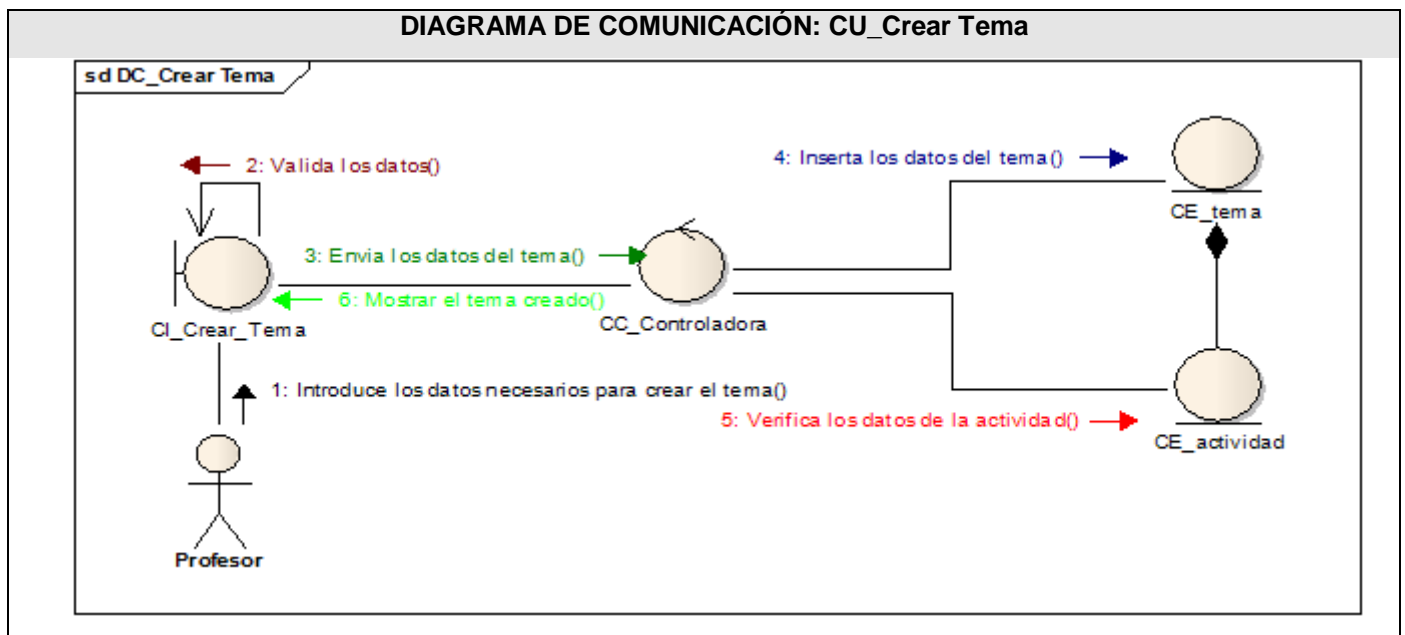


Figura 10. Diagrama de comunicación Crear Tema (elaboración propia)

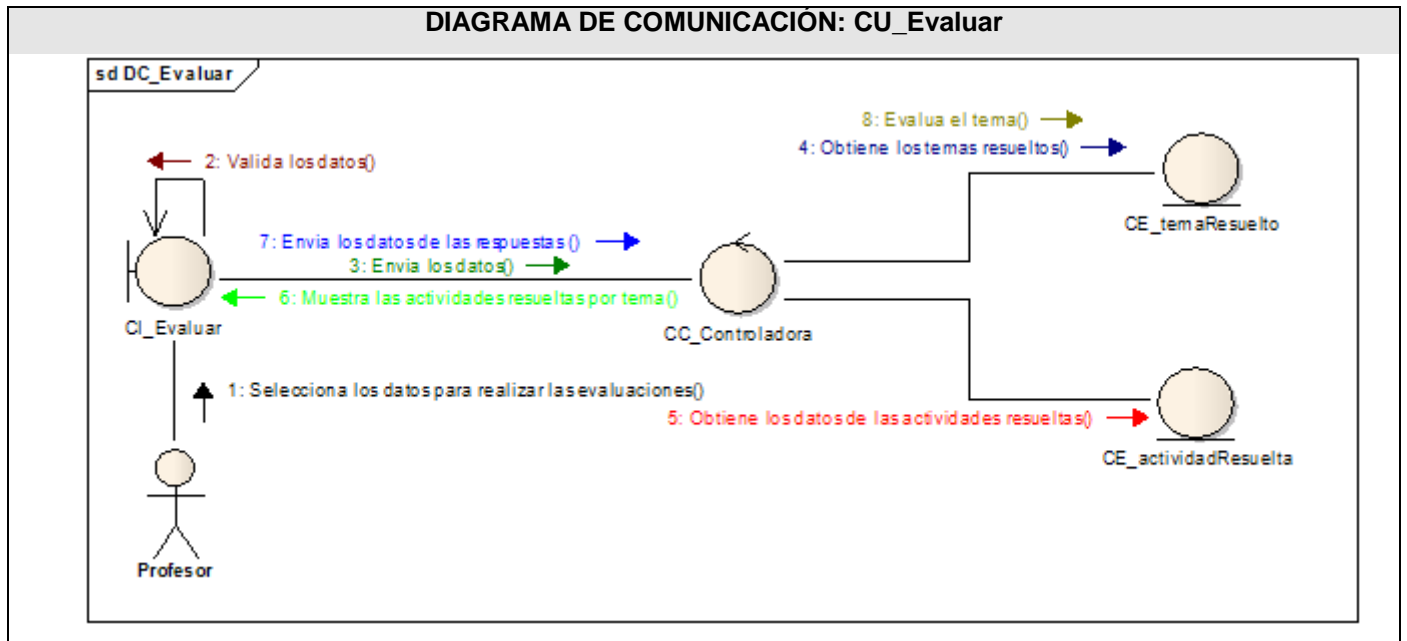


Figura 11. Diagrama de comunicación Evaluar (elaboración propia)

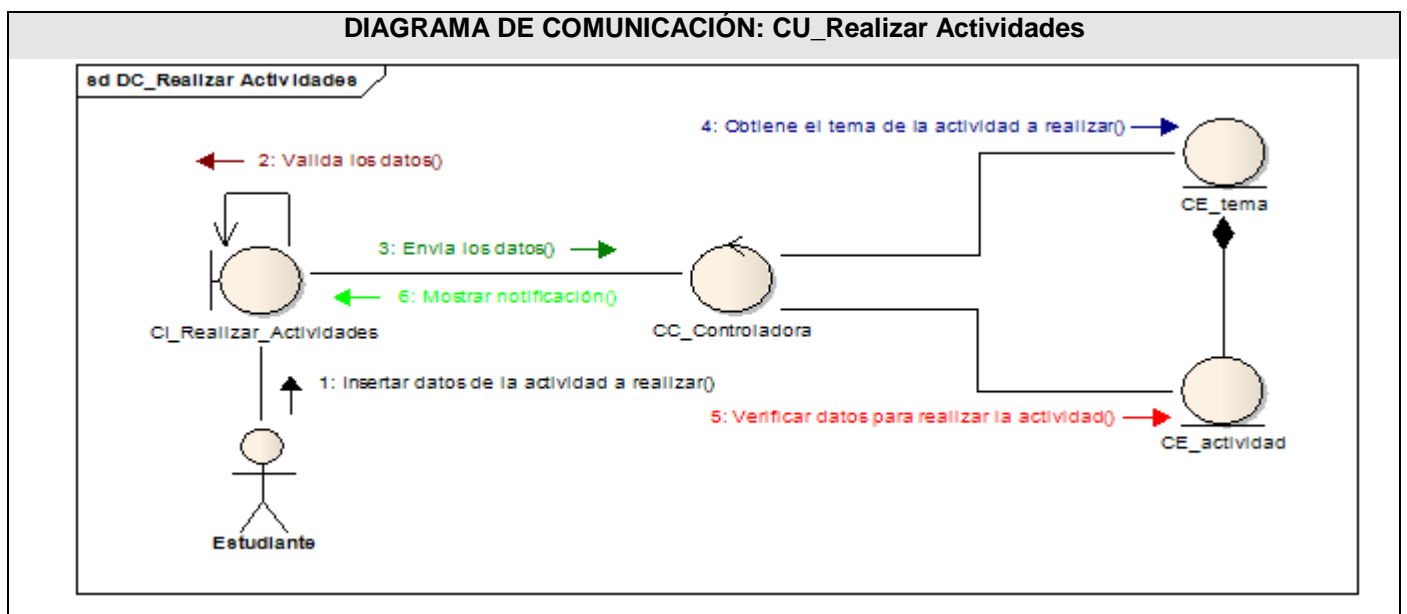


Figura 12. Diagrama de comunicación Realizar Actividades (elaboración propia)

### **3.2 Modelo de diseño**

El modelo de diseño, es un modelo de objetos que describe la realización física de los casos de uso, centrándose en cómo los requisitos funcionales y no funcionales junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar. Además, el modelo sirve de abstracción de la implementación del sistema y es, de ese modo, utilizada como una entrada fundamental de las actividades de implementación. (Hidalgo, y otros, 2010)

#### **3.2.1 Diagramas de clases del diseño**

Las clases de análisis siempre encajan en uno de tres estereotipos básicos: de interfaz, de control o de entidad. Cada estereotipo implica una semántica específica que constituye un método potente y consistente de identificar y describir las clases de análisis. (Jacobson, 2000)

A continuación se representan los diagramas de clase del diseño de los casos de uso arquitectónicamente significativos.

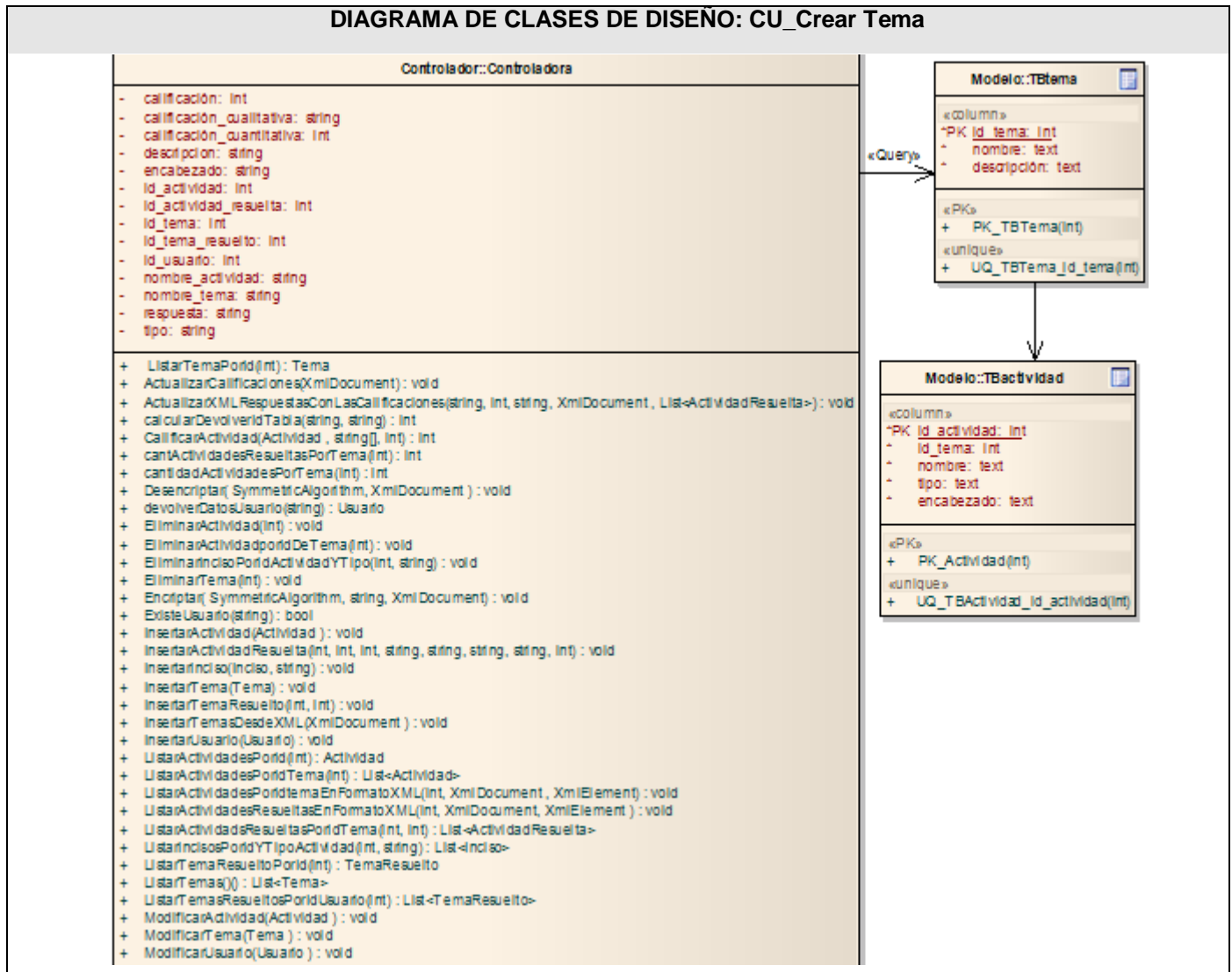


Figura 13. Diagrama de clases de diseño Crear Tema (elaboración propia)



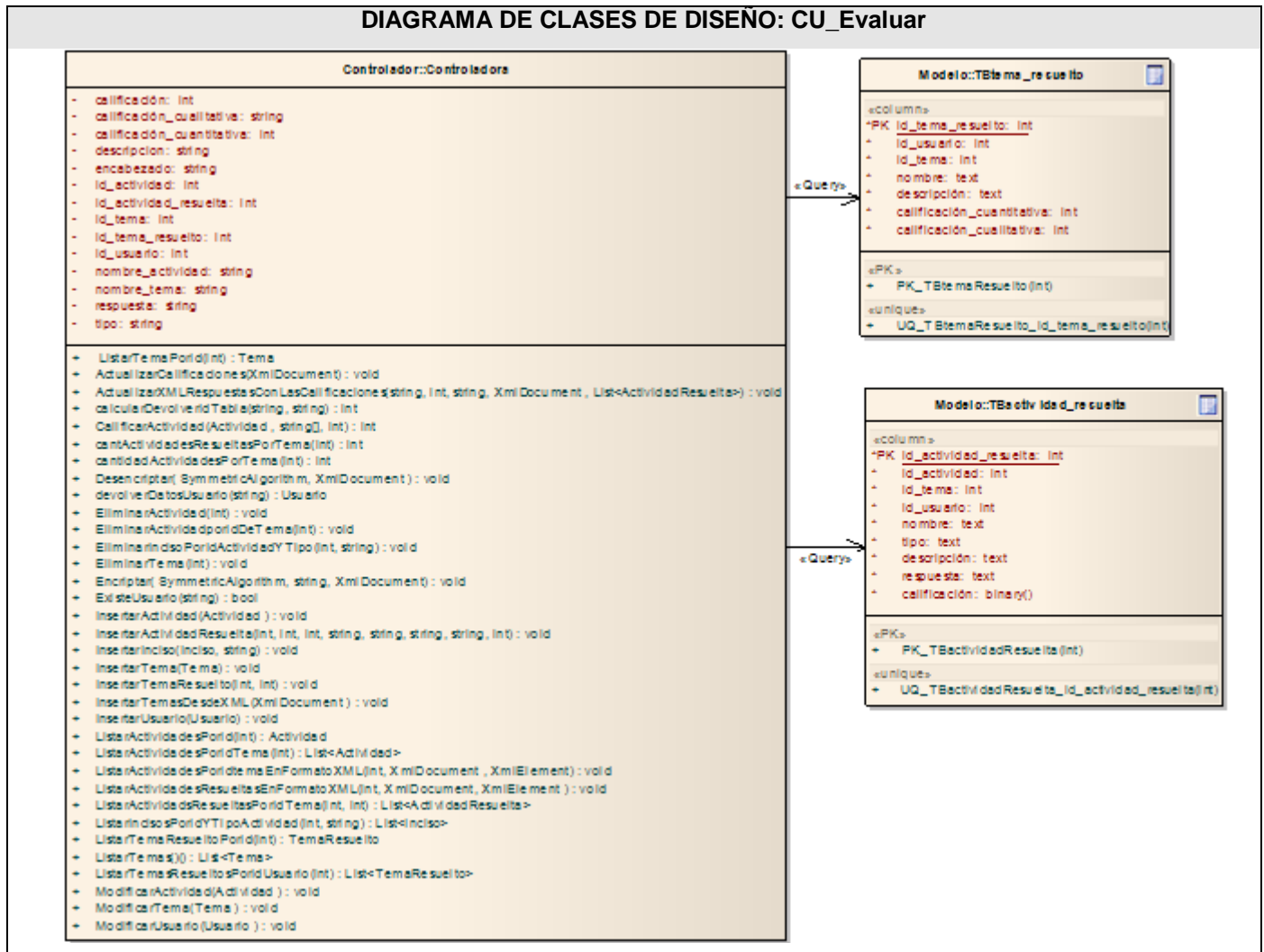


Figura 14. Diagrama de clases de diseño Evaluar (elaboración propia)

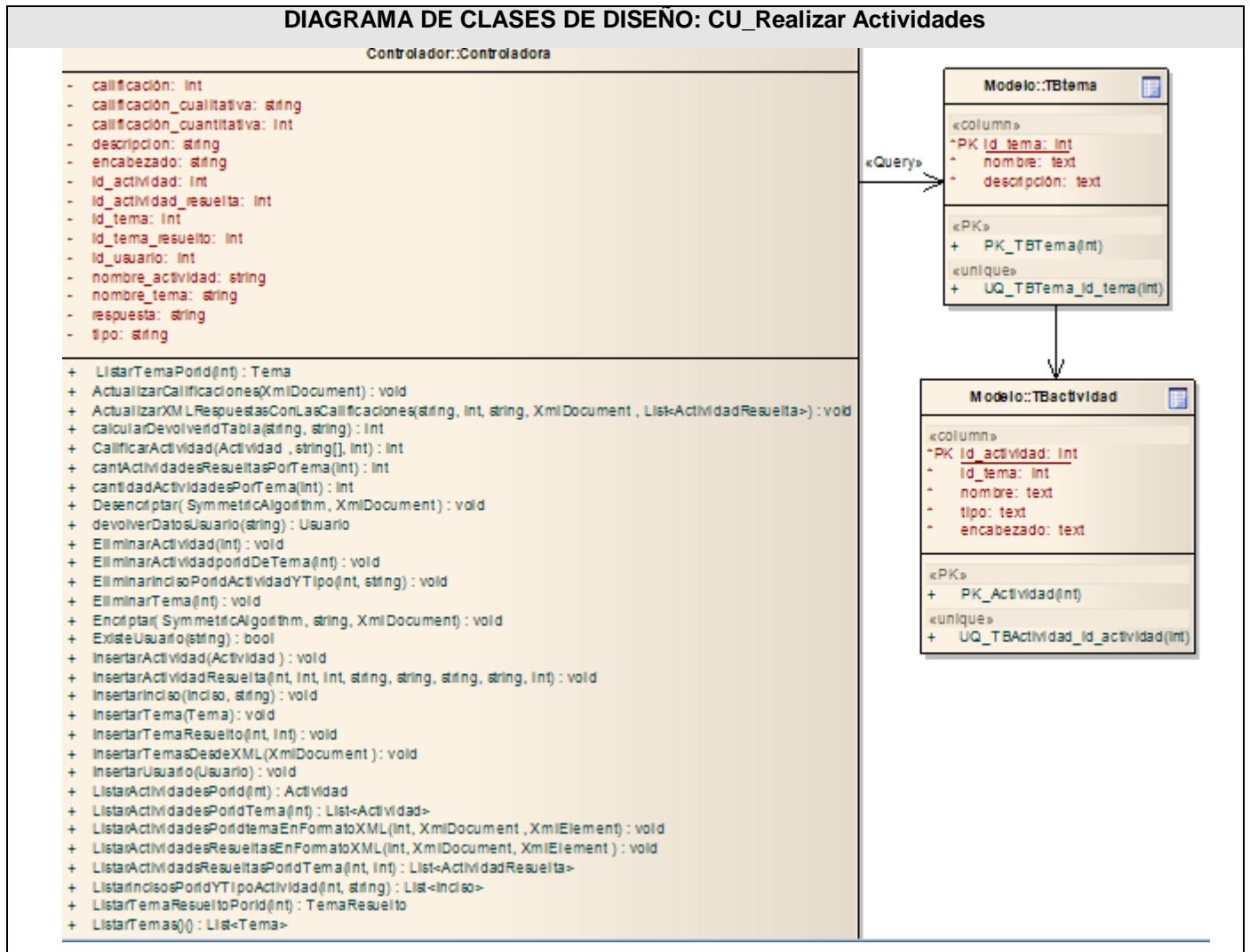


Figura 15. Diagrama de clases de diseño Realizar Actividades (elaboración propia)

### 3.2.1.1 Descripción de las clases

A continuación se realiza una descripción de alguna de las clases del modelo de diseño.

Tabla 8. Descripción de la clase controladora (elaboración propia)

|                                    |  |
|------------------------------------|--|
| <b>Nombre:</b> Controladora        |  |
| <b>Tipo de clase:</b> Controladora |  |
| <b>Atributo</b>                    | <b>Tipo</b>  |
| id_tema                            | int  |
| id_actividad                       | int  |
| id_actividad_resuelta              | int  |
| id_tema_resuelto                   | int  |
| id_usuario                         | int  |
| Calificación                       | int  |
| calificación cuantitativa          | int  |
| calificación cualitativa           | string   |
| Descripción                        | string   |
| Encabezado                         | string   |
| nombre_actividad                   | string   |
| nombre_tema                        | string   |
| Respuesta                          | string   |
| Tipo                               | string   |
| <b>Para cada responsabilidad:</b>  |  |
| <b>Nombre:</b>                     | ExisteUsuario(string usuarioParam)   |
| <b>Descripción:</b>                | Verifica si existe un usuario o no en la aplicación  |
| <b>Nombre</b>                      | devolverDatosUsuario (string usuarioParam)   |
| <b>Descripción</b>                 | Devuelve un objeto con los datos del usuario que tenga el atributo usuario igual al pasado por parámetro |
| <b>Nombre</b>                      | InsertarUsuario (Usuario u)  |
| <b>Descripción</b>                 | Adiciona un nuevo usuario a la aplicación  |
| <b>Nombre</b>                      | calcularDevolverIdTabla (string nombreTabla, string nombreColumna)                                       |
| <b>Descripción</b>                 | Devuelve el mayo id de la tabla especificada incrementado en 1   |
| <b>Nombre</b>                      | InsertarTema (Tema t)  |

### Capítulo 3: Análisis y diseño e implementación del sistema SAI

|                    |   |
|--------------------|---|
| <b>Descripción</b> | Adiciona un nuevo tema a la aplicación  |
| <b>Nombre</b>      | InsertarActividad (Actividad A)   |
| <b>Descripción</b> | Adiciona una nueva actividad al tema  |
| <b>Nombre</b>      | InsertarInciso(Inciso I, string tipo)   |
| <b>Descripción</b> | Adiciona un nuevo inciso a la actividad   |
| <b>Nombre</b>      | InsertarTemasDesdeXML(XmlDocument archivoxml)   |
| <b>Descripción</b> | Adiciona en la base de datos los temas contenidos en archivo XML  |
| <b>Nombre</b>      | EliminarActividad(int id)   |
| <b>Descripción</b> | Elimina una actividad de la aplicación dado su id   |
| <b>Nombre</b>      | EliminarIncisoPorIdActividadYTipo(int id_actividad, string tipo)  |
| <b>Descripción</b> | Elimina un inciso de una actividad específica dado su id y su tipo  |
| <b>Nombre</b>      | EliminarActividadporIdDeTema(int id_tema)   |
| <b>Descripción</b> | Elimina una actividad dado el id del tema   |
| <b>Nombre</b>      | EliminarTema(int id)  |
| <b>Descripción</b> | Elimina un tema de la aplicación  |
| <b>Nombre</b>      | ListarTemas()   |
| <b>Descripción</b> | Muestra un listado con los temas  |
| <b>Nombre</b>      | ListarTemaPorId(int id_tema)  |
| <b>Descripción</b> | Devuelve el tema cuyo id sea igual al pasado por parámetro  |
| <b>Nombre</b>      | ListarActividadesPorId(int id_actividad)  |
| <b>Descripción</b> | Devuelve la actividad cuyo id sea igual al pasado por parámetro   |
| <b>Nombre</b>      | ListarActividadesPorIdTema(int id_tema)   |
| <b>Descripción</b> | Muestra un listado con las actividades cuyo id del tema sea igual al pasado por parámetro                           |
| <b>Nombre</b>      | ListarIncisosPorIdYTipoActividad(int id_actividad, string tipo)   |
| <b>Descripción</b> | Muestra un listado con los incisos de un tipo de actividad cuyo id de la actividad se igual al pasado por parámetro |
| <b>Nombre</b>      | ModificarUsuario(Usuario u)   |
| <b>Descripción</b> | Modifica un usuario existente   |

### Capítulo 3: Análisis y diseño e implementación del sistema SAI

|                    |   |
|--------------------|---|
| <b>Nombre</b>      | ModificarActividad(Actividad A)   |
| <b>Descripción</b> | Modifica una actividad dada   |
| <b>Nombre</b>      | ModificarTema(Tema T)   |
| <b>Descripción</b> | Modifica un tema dado   |
| <b>Nombre</b>      | ListarActividadesPorIdtemaEnFormatoXML(int id_temaParametro, XmlDocument doc, XmlElement datos)   |
| <b>Descripción</b> | Lista las actividades en formato XML donde el id del tema sea igual al pasado por parámetro   |
| <b>Nombre</b>      | ListarActividadesResueltasEnFormatoXML(int id_temaParametro, XmlDocument doc, XmlElement data)  |
| <b>Descripción</b> | Lista las actividades resueltas en formato XML  |
| <b>Nombre</b>      | CalificarActividad(Actividad A, string[] respuestasEstudiante, int id_usuario)  |
| <b>Descripción</b> | Realiza la calificación de una actividad realizada por un estudiante  |
| <b>Nombre</b>      | InsertarActividadResuelta(int id_actividad, int id_tema, int id_usuario, string nombre, string tipo, string encabezado, string respuesta, int calificacion) |
| <b>Descripción</b> | Adiciona una actividad resuelta de un tema específico   |
| <b>Nombre</b>      | InsertarTemaResuelto(int id_tema, int id_usuario)   |
| <b>Descripción</b> | Adiciona un tema resuelto por un usuario  |
| <b>Nombre</b>      | cantidadActividadesPorTema(int id_tema)   |
| <b>Descripción</b> | Devuelve la cantidad de actividades de un tema en específico  |
| <b>Nombre</b>      | cantActividadesResueltasPorTema(int id_tema)  |
| <b>Descripción</b> | Devuelve la cantidad de actividades resueltas de un tema específico   |
| <b>Nombre</b>      | ListarTemasResueltosPorIdUsuario(int id_usuarioParam)   |
| <b>Descripción</b> | Muestra una lista con los temas resueltos por un usuario dado   |
| <b>Nombre</b>      | ListarTemaResueltoPorId(int id)   |
| <b>Descripción</b> | Muestra el tema resuelto cuyo id se igual al pasado por parámetro   |
| <b>Nombre</b>      | ListarActivadsResueltasPorIdTema(int id_temaParam, int id_usuarioParam)   |
| <b>Descripción</b> | Muestra una lista con las actividades resueltas de un tema dado   |
| <b>Nombre</b>      | ActualizarXMLRespuestasConLasCalificaciones(string id_temaParam, int  |

|                    |  |
|--------------------|--|
|                    | calificacion_cuant,string calificacion_cualit, List<ActividadResuelta> actividades_calificadas, XmlDocument archivo) |
| <b>Descripción</b> | Actualiza el XML con las calificaciones de las respuestas dadas por un usuario                                       |
| <b>Nombre</b>      | ActualizarCalificaciones(XmlDocument archivo)  |
| <b>Descripción</b> | Actualiza en la base de datos las calificaciones de un usuario desde un archivo XML                                  |
| <b>Nombre</b>      | Encriptar(XmlDocument Doc, string ElementName, SymmetricAlgorithm Key)   |
| <b>Descripción</b> | Encripta los datos del XML dado con la llave dada  |
| <b>Nombre</b>      | Desencriptar(XmlDocument Doc, SymmetricAlgorithm Alg)  |
| <b>Descripción</b> | Desencripta los datos del XML dado con la llave dada   |

Tabla 9. Descripción de la clase TBtema (elaboración propia)

|                               |             |
|-------------------------------|-------------|
| <b>Nombre:</b> TBtema         |             |
| <b>Tipo de clase:</b> Entidad |             |
| <b>Atributo</b>               | <b>Tipo</b> |
| id_tema                       | int         |
| Nombre                        | texto       |
| Descripción                   | texto       |

Tabla 10. Descripción de la clase TBactividad (elaboración propia)

|                               |             |
|-------------------------------|-------------|
| <b>Nombre:</b> TBactividad    |             |
| <b>Tipo de clase:</b> Entidad |             |
| <b>Atributo</b>               | <b>Tipo</b> |
| id_actividad                  | int         |
| id_tema                       | int         |
| Nombre                        | texto       |
| Tipo                          | text        |
| Encabezado                    | texto       |

### 3.2.2 Modelo de datos

Un modelo de datos, es un conjunto de conceptos que permiten describir los datos, las relaciones que existen entre ellos, la semántica y las restricciones de consistencia. (Rivera, 2010)

A continuación se muestra el modelo de base de datos.

**Leyenda:** La tabla que se encuentra en color amarillo, es una tabla adicional para cuando se trabaja con el servidor de base de datos Postgre.

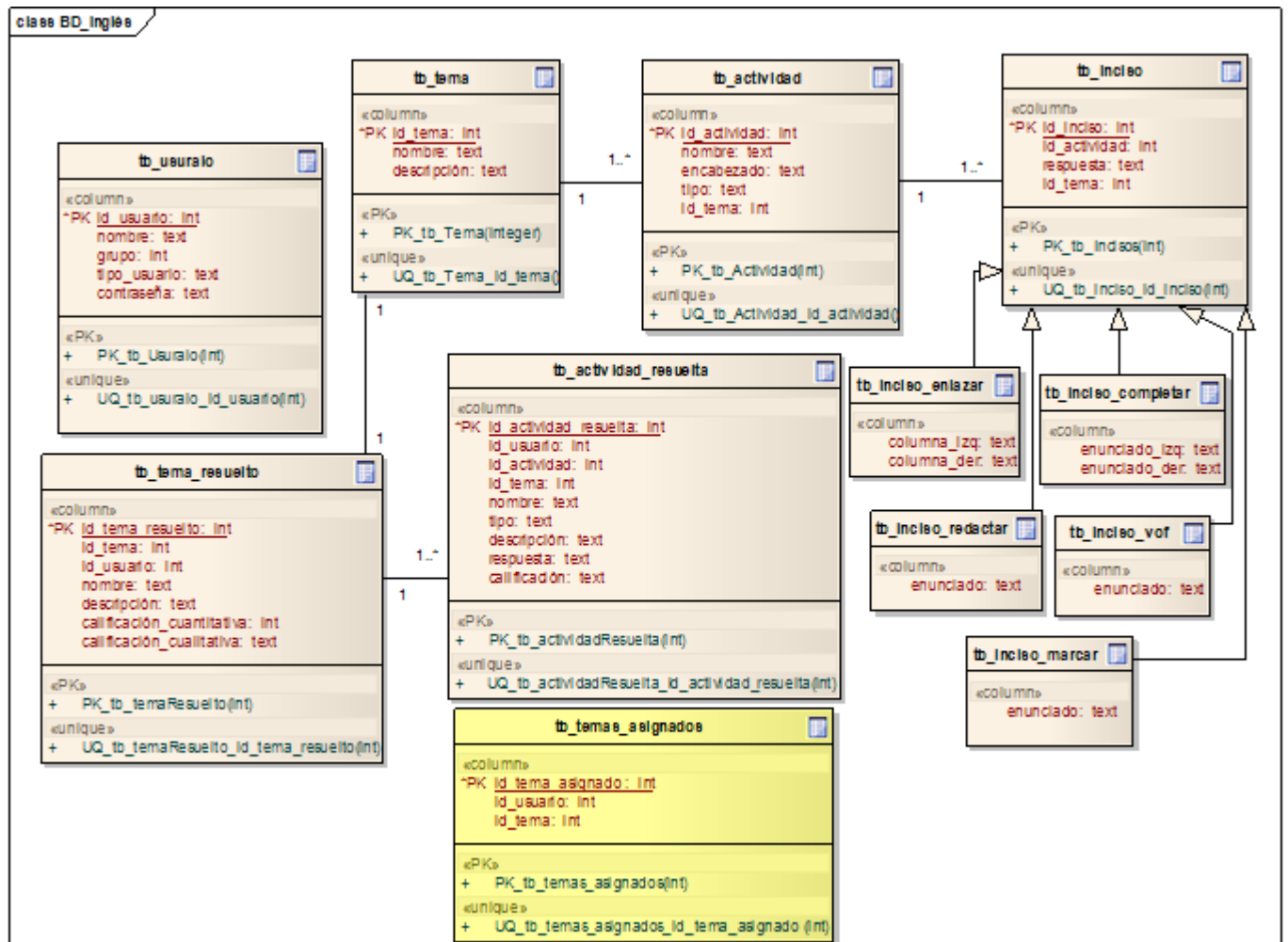


Figura 16. Modelo de base de datos (elaboración propia)

### 3.2.2.1 Descripción de algunas de las tablas de la base de datos

A continuación se realiza una descripción de algunas de las tablas del modelo de base de datos:

Tabla 11. Descripción de la tabla tb\_usuario (elaboración propia)

| <b>Nombre: tb_usuario</b>   |      |   |
|---|------|---|
| <b>Descripción:</b> Guarda toda la información de los usuarios que interactúan con la aplicación. |      |   |
| Atributo  | Tipo | Descripción   |
| Id_usuario  | int  | Identificador único para el usuario.                  |
| Nombre  | text | Nombre del usuario que se autentica en la aplicación. |
| Grupo   | int  | Grupo al que pertenece el usuario.                    |
| tipo_usuario  | text | Tipo que es el usuario.                               |
| Contraseña  | text | Clave propia del usuario.                             |

Tabla 12. Descripción de la tabla tb\_tema (elaboración propia)

| <b>Nombre: tb_tema</b>  |      |                                   |
|---|------|-----------------------------------|
| <b>Descripción:</b> Almacena información referente a los temas. |      |                                   |
| Atributo  | Tipo | Descripción                       |
| id_tema   | Int  | Identificador único para el tema. |
| Nombre  | Text | Nombre del tema.                  |
| Descripción   | Text | Descripción del tema.             |

Tabla 13. Descripción de tb\_temaResuelto (elaboración propia)

| <b>Nombre: tb_temaResuelto</b>  |      |  |
|---|------|--|
| <b>Descripción:</b> Almacena información referente a los temas resueltos. |      |  |
| Atributo  | Tipo | Descripción  |
| id_tema_resuelto  | int  | Identificador único del tema resuelto.               |
| id_usuario  | int  | Identificador único del usuario que realizó el tema. |
| id_tema   | int  | Identificador para buscar el tema resuelto.          |



|                           |      |  |
|---------------------------|------|--|
| Nombre                    | text | Nombre del tema resuelto.                                |
| Descripción               | text | Descripción del tema resuelto.                           |
| calificación cuantitativa | int  | Tipo de calificación que se le da a los temas resueltos. |
| calificación cualitativa  | text | Tipo de calificación que se le da a los temas resueltos. |

### 3.3 Modelo de implementación

El modelo de implementación describe cómo los elementos del modelo de diseño se implementan en términos de componentes. Describe también, cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación, y en el lenguaje o lenguajes de programación utilizados y cómo dependen los componentes unos de otros. (Jacobson, 2004)

#### 3.3.1 Diagrama de componentes

Los diagramas de componentes ilustran las piezas del software y controladores embebidos que conformarán un sistema. Un diagrama de componentes tiene un nivel más alto de abstracción que un diagrama de clase, usualmente un componente se implementa por una o más clases (u objetos) en tiempo de ejecución. Estos son bloques de construcción, y eventualmente pueden comprender una gran porción de un sistema. (Sparxsystems, 2007b)

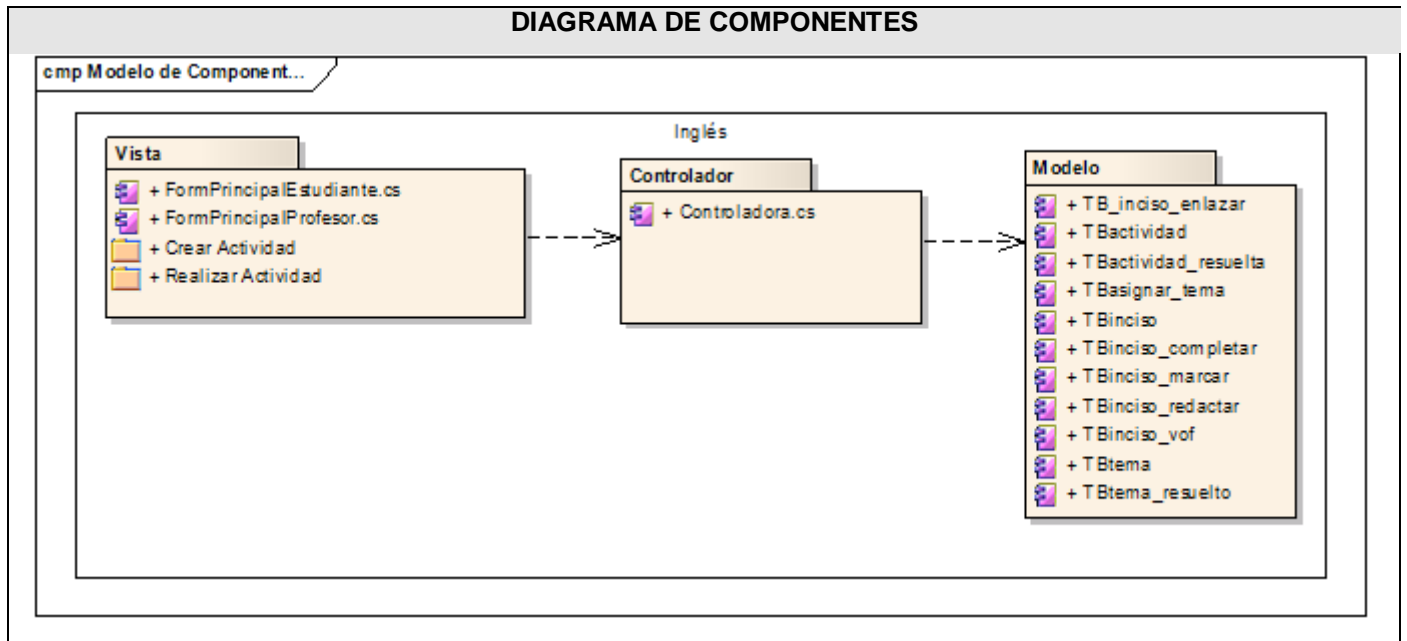


Figura 17. Diagrama de componentes general (elaboración propia)

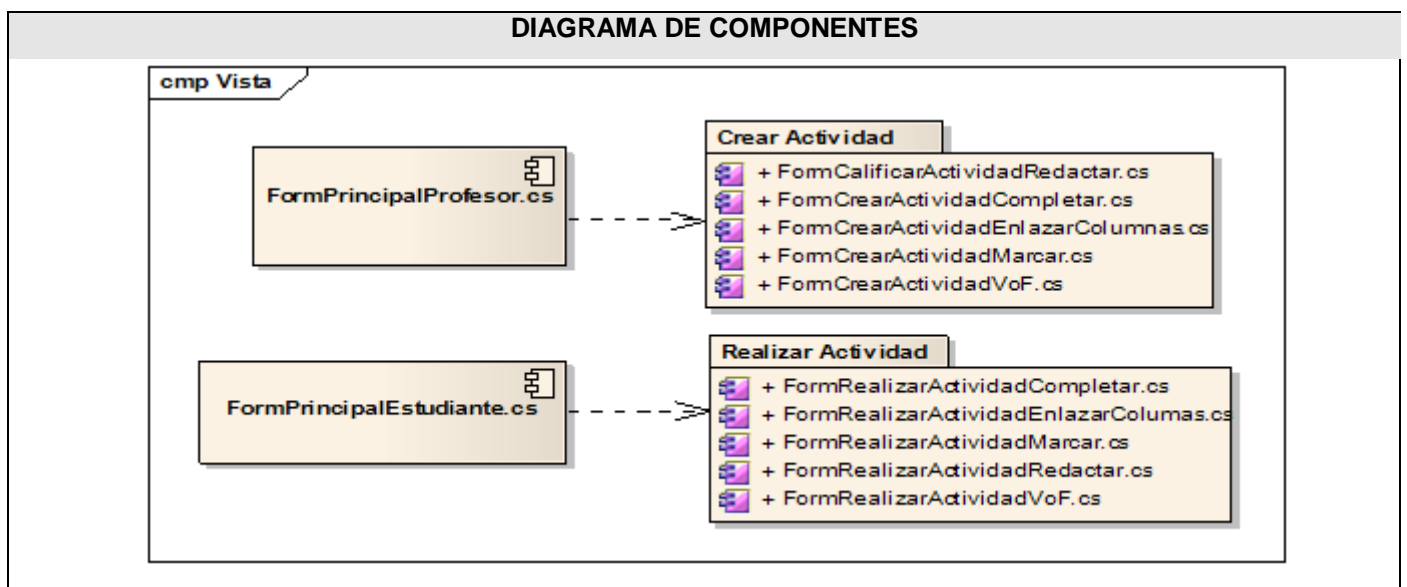


Figura 18. Diagrama de componentes Vista (elaboración propia)

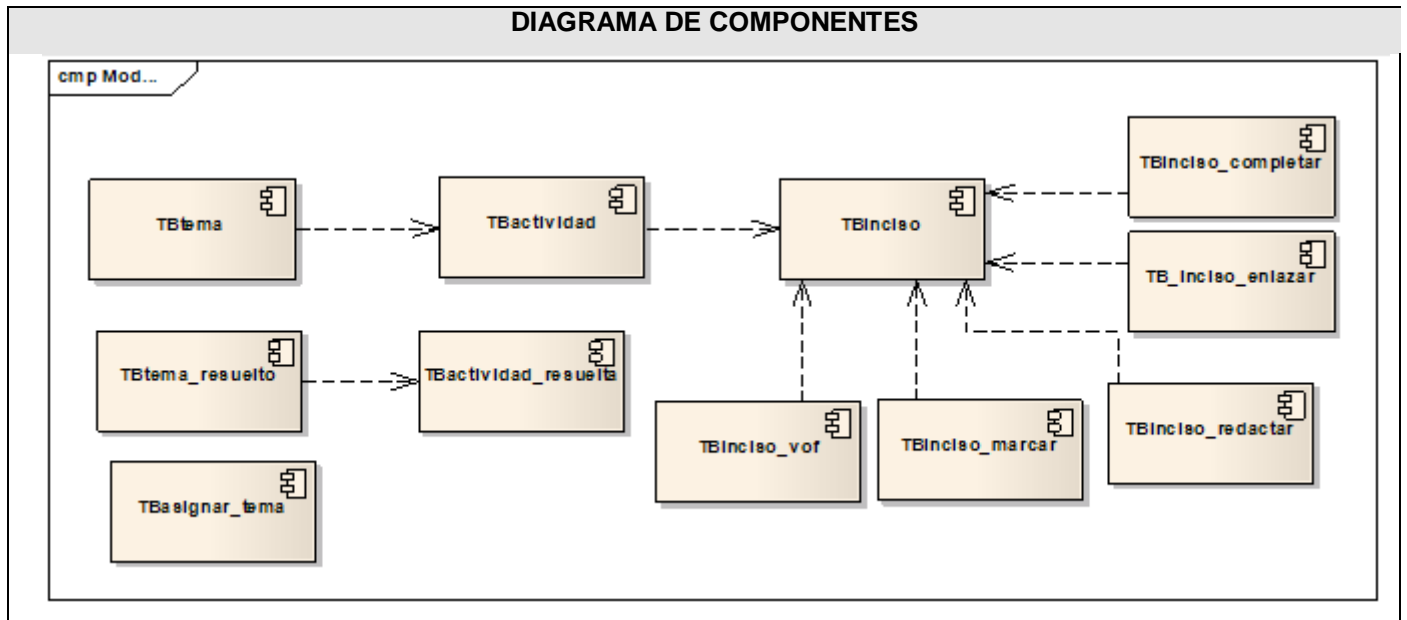


Figura 19. Diagrama de componentes Modelo (elaboración propia)

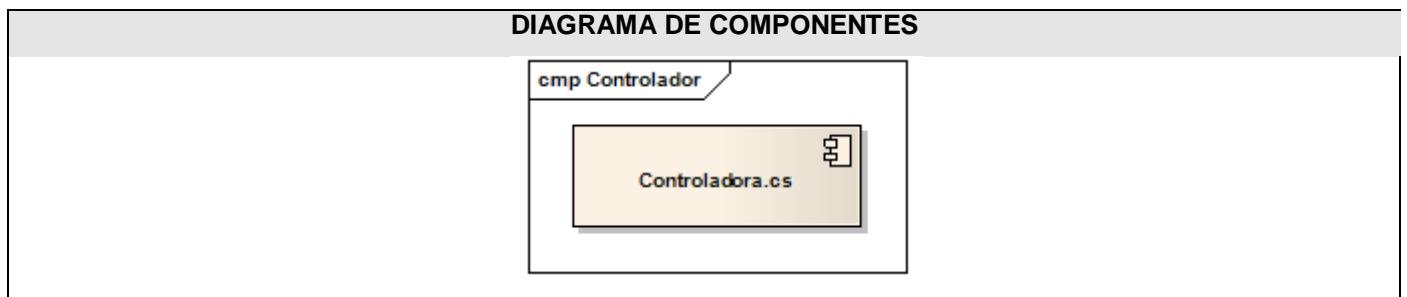


Figura 20. Diagrama de componentes Controlador (elaboración propia)

### 3.3.2 Diagrama de despliegue

Un diagrama de despliegue muestra cómo y dónde se desplegará el sistema. Las máquinas físicas y los procesadores se representan como nodos, y la construcción interna puede ser representada por nodos o artefactos embebidos. Como los artefactos se ubican en los nodos para modelar el despliegue del sistema, la ubicación es guiada por el uso de las especificaciones de despliegue. (Sparxsystems, 2007a) A continuación se muestra el diagrama de despliegue de la aplicación cuando se utiliza la conexión a la base de datos interna.

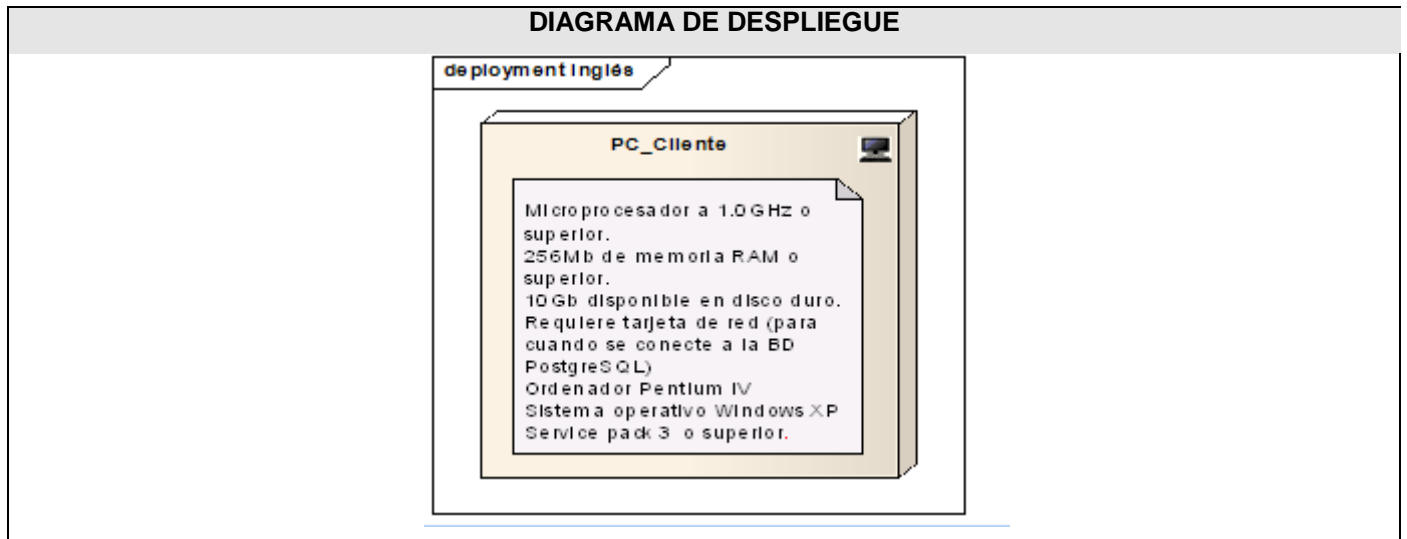


Figura 21. Diagrama de despliegue (elaboración propia)

A continuación se muestra el diagrama de despliegue de la aplicación, cuando se utiliza la conexión a la base de datos remota. En este caso, la aplicación se instala en la PC cliente de donde se conecta al servidor mediante el protocolo TCP/IP.

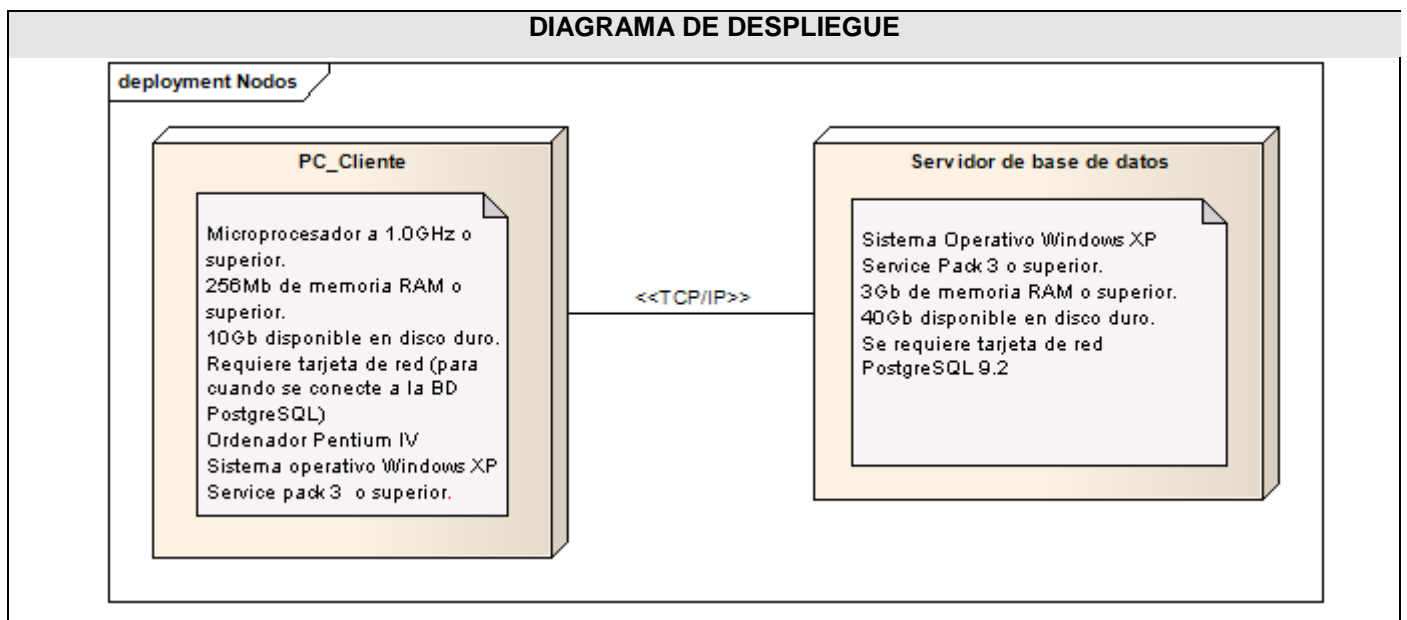


Figura 22. Diagrama de despliegue (elaboración propia)

### **3.4 Seguridad**

Para la seguridad de los ficheros XML que se usan en la aplicación, se utilizó el algoritmo de cifrado Advanced Encryption Standard (AES), también conocido como Rijndael, el mismo es un algoritmo simétrico. Por ser simétrico, se utiliza la misma clave para encriptar como para desencriptar y la longitud de la clave puede ser de 128, 192 o 256 bits según especifica el estándar. El proceso de cifrado del algoritmo consiste, en aplicar a cada estado un conjunto de operaciones agrupadas a las que se le denominan rondas; el algoritmo realiza 11 rondas, donde en cada ronda se aplica una subclave diferente. El proceso de descifrado aplica las mismas operaciones que el cifrado pero de forma inversa utilizando las mismas subclaves generadas en orden inverso. (Pousa, 2011)

Para hacer uso de este algoritmo en la aplicación, se utiliza la clase RijndaelManaged de Monodevelop, la cual contiene varias funciones para cifrar y descifrar datos con AES. También se utiliza la clase EncryptedXml, que es la que permite en Monodevelop, que un archivo XML o parte de sus datos puedan ser cifrados. El uso de AES en la aplicación garantiza que no se acceda a la información contenida en dichos ficheros desde otro programa que no sea la aplicación. Esto evita, que los estudiantes tengan acceso a las respuestas de las actividades contenidas en los temas, que se guardan en estos archivos antes de resolverlas. También impide que los mismos modifiquen las calificaciones obtenidas en las actividades para otorgarse una mejor.

### Conclusiones

Al concluir la investigación en curso se concluye que:

- Los sistemas informáticos relacionados con el aprendizaje del idioma Inglés analizados, no brindan una solución al problema planteado, debido que no presentan todas las características que se necesitan abordar en la solución.
- Se realizó un análisis, de los temas de la producción que podrían ser de interés para los estudiantes; identificando que los temas más estudiados son: tecnologías, herramientas y lenguajes de programación.
- La metodología, tecnologías y herramientas definidas para el desarrollo de la aplicación, constituyeron los elementos necesarios para fundamentar las bases teóricas de la solución obtenida.

### **Recomendaciones**

Al finalizar la presente investigación se recomienda:

- Permitir que desde la aplicación, el profesor pueda asignar distintos valores a los incisos de las actividades, según la nota que estime.
- Permitir incorporar, por el profesor, más de diez incisos en cada actividad de forma dinámica.
- Implementar funcionalidades a la aplicación, para que la misma, sea capaz de asignar temas a los estudiantes automáticamente, según los resultados obtenidos por estos y la complejidad de los temas.

Referencias bibliográficas

1. (Arrocha, 2008) **Arrocha, Juan Rafael Curbelo. 2008.** neleste.com. [En línea] 22 de 07 de 2008. [Citado el: 20 de 02 de 2013.] <http://www.neleste.com/modelo-vista-controlador/> .
2. (Brown, 2007) **Brown. 2007.** ec.europa.eu. [En línea] 2007. [Citado el: 16 de 11 de 2012.] [http://ec.europa.eu/languages/documents/publications/learn\\_es.pdf](http://ec.europa.eu/languages/documents/publications/learn_es.pdf).
3. (Carretero, y otros, 2008) **Carretero, Valentín y Marín, Omar. 2008.** [En línea] 2008. [Citado el: 07 de 06 de 2013.] [http://iessanvicente.com/colaboraciones/berkeley\\_interbase.pdf](http://iessanvicente.com/colaboraciones/berkeley_interbase.pdf).
4. (Craig, 1999) **Craig, Larman. 1999.** *UML y Patrones*. s.l. : Segunda Edición ed. Prentice Hall, 1999.
5. (Cubaeduca, 2009) **Cubaeduca. 2009.** cubaeduca.cu. *Portal Educativo Cubano*. [En línea] 12 de 11 de 2009. [Citado el: 28 de 02 de 2013.] [http://www.cubaeduca.cu/index.php?option=com\\_content&view=article&id=4690&Itemid=153](http://www.cubaeduca.cu/index.php?option=com_content&view=article&id=4690&Itemid=153).
6. (Dwyer, 2010) **Dwyer, Owen. 2010.** educacion-con-tic.blogspot.com. [En línea] 2010. [Citado el: 20 de 12 de 2012.] <http://educacion-con-tic.blogspot.com/2013/01/juegos-para-aprender-ingles.html>.
7. (Equipo de Softonic , 2011 ) **Equipo de Softonic . 2011 .** sqlitemanager.softonic.com. [En línea] 05 de 08 de 2011 . [Citado el: 20 de 04 de 2013.] <http://sqlitemanager.softonic.com/> .
8. (Ferguson, y otros, 2003 pág. 37) **Ferguson, Jeff, Patterson, Brian y Beres, Jason. 2003.** *La biblia de C#*. [ed.] 15.28027 Juan Ignacio Luca de Tena. Madrid : EDICIONES ANAYA MULTIMEDIA (GRUPO ANAYA, S.A.), 2003. pág. 37. <http://bibliodoc.uci.cu/pdf/8441514844.pdf>.
9. (Ferrera, 2010) **La Tecnología informática en la sociedad cubana. Ferrera, Miriam Cecilia Gómez y Domínguez, Joel Fuentes. 2010.** [ed.] Juana Ivis Imamura Díaz. Ciudad de la Habana : s.n., 2010, Pedagogía Profesional, Vols. Edición especial,2. <http://www.pedagogiaprofesional.rimed.cu/especial%202/Joe.htm>. ISSN 1684-5765.
10. (Figueroa, y otros, 2011) **Figueroa, Roberth G., Solís, Camilo J y Cabrera, Armando A. 2011.** *Metodologías tradicionales VS. Metodologías ágiles*. 2011. pág. 2. [http://eva.uci.cu/file.php/161/Documentos/Materiales\\_complementarios/UD\\_1\\_Procesos/Metodologias/METODOLOGIAS\\_TRADICIONALES\\_VS.\\_METODOLOGIAS\\_AGILES.pdf](http://eva.uci.cu/file.php/161/Documentos/Materiales_complementarios/UD_1_Procesos/Metodologias/METODOLOGIAS_TRADICIONALES_VS._METODOLOGIAS_AGILES.pdf).
11. (GSInnova, 2012) **GSInnova. 2012.** rational.com.ar. [En línea] 2012. [Citado el: 20 de 02 de 2013.] <http://www.rational.com.ar/herramientas/rup.html>.
12. (Guia-ubuntu, 2008) —. **2008.** guia-ubuntu. [En línea] 10 de 03 de 2008. [Citado el: 26 de 05 de 2013.] <http://www.guia->



13. (Guia-ubuntu, 2011) **Guia-ubuntu. 2011.** guia-ubuntu. [En línea] 29 de julio de 2011. [Citado el: 10 de 05 de 2013.] [http://www.guia-ubuntu.com/index.php?title=PgAdmin\\_III](http://www.guia-ubuntu.com/index.php?title=PgAdmin_III).
14. (Hidalgo, y otros, 2010) **Hidalgo, Daira Figueroa, Ortiz, Yurisbel Vega y Fernández, Vladimir Martell. 2010.** Propuesta de diseño para proyectos informáticos que utilizan Symfony como Framework. [En línea] 2010. [Citado el: 20 de 05 de 2013.] [publicaciones.uci.cu/index.php/SC/article/download/208/228](http://publicaciones.uci.cu/index.php/SC/article/download/208/228).
15. (Hipp, 2011) **Hipp, D. Ri-chard. 2011.** *sqlite-latino.blogspot.com*. [En línea] 12 de 04 de 2011. [Citado el: 05 de 03 de 2013.] <http://sqlite-latino.blogspot.com/2011/04/sqlite-articulo-en-pdf.html>.
16. (InstEd, 2009 ) **InstEd. 2009** . *blogs.rimed.cu*. [En línea] 12 de 11 de 2009 . [Citado el: 28 de 02 de 2013.] <http://blogs.rimed.cu/futuro/files/2009/11/Manual-Sunrise.pdf> .
17. (Jacobson, 2000) **Jacobson, James. 2000.** *El proceso unificado de desarrollo de software*. Madrid : Pearson Educación, S.A, 2000.
18. (Jacobson, 2004) —. **2004.** *El proceso unificado de desarrollo de software*. [ed.] Addison Wesley. 2004.
19. (Java, 2009) **Java. 2009.** *lenguajes-de-programacion.com*. *Lenguajes de programación*. [En línea] 2009. [Citado el: 20 de 11 de 2013.] <http://www.lenguajes-de-programacion.com/programacion-java.shtml>.
20. (León, 2010 pág. 12) *Entornos Virtuales de Aprendizaje: Herramienta fundamental para desarrollar con mayor efectividad los procesos*. **León, Yordanis Medina y Ramírez, Yaneisy Castilla. 2010.** 1, La Habana : s.n., 2010, Serie Científica, Universidad de las Ciencias Informáticas, Vol. 3, pág. 12. <http://publicaciones.uci.cu/index.php/SC/issue/view/28>. ISSN: 2306-2495.
21. (Márquez, 2007 p. 3) *Evaluación del desempeño para entorno virtual de autoaprendizaje de Inglés en la Universidad de*. **Márquez, Yoan Martínez y Batista, Yalice Gámez. 2007.** 1, 2007, Serie Científica, Universidad de las Ciencias Informáticas, Vol. 1. ISSN: 2306-2495.
22. (Martínez, 2009) **Martínez, Prof. Ivette Carolina. 2009.** Clase 6: Modelo conceptual/Modelo de dominio. [En línea] 2009. [Citado el: 10 de 03 de 2013.] <http://www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r88846.PDF>.
23. (MongoDB, 2010) **MongoDB. 2010.** *mongoDB*. [En línea] 2010. [Citado el: 07 de 06 de 2013.] <http://www.mongodb.org>.
24. (Monodevelop, 2010) **Monodevelop. 2010.** *monodevelop.com*. [En línea] 6 de 12 de 2010. [Citado el: 20 de 2 de 2013.] <http://monodevelop.com/> .
25. (Montiel, 2008) **Montiel, Daniel Ponsoda. 2008.** Introducción a SQLite. [En línea] 23 de enero de 2008. [Citado el: 06 de 06 de 2013.]

26. (Nuryr, 2008) <https://iessanvicente.com/colaboraciones/sqlite>.  
**Nuryr. 2008.** tic\_nury.zoomblog.com. [En línea] 31 de 10 de 2008. [Citado el: 20 de 11 de 2012.]  
[http://tic\\_nury.zoomblog.com/archivo/2008/10/31/importancia-de-las-Tics-para-el-Desarr.html](http://tic_nury.zoomblog.com/archivo/2008/10/31/importancia-de-las-Tics-para-el-Desarr.html).
27. (Orallo, 2007 pág. 1) **Orallo, Enrique Hernández. 2007.** *El Lenguaje Unificado de Modelado(UML)*. España : Valencia s.n, 2007.
28. (Postgresql, 2010) **Postgresql. 2010.** postgresql. [En línea] 10 de 10 de 2010. [Citado el: 21 de 01 de 2013.] <http://www.postgresql.org/about/>.
29. (Pousa, 2011) **Pousa, Lic. Adrián. 2011.** ALGORITMO DE CIFRADO SIMÉTRICO AES. ACELERACIÓN DE TIEMPO DE CÓMPUTO SOBRE ARQUITECTURAS MULTICORE. [En línea] diciembre de 2011. [Citado el: 20 de 05 de 2013.]  
[http://postgrado.info.unlp.edu.ar/Carreras/Especializaciones/Redes\\_y\\_Seguridad/Trabajos\\_Finales/Pousa\\_Adrian.pdf](http://postgrado.info.unlp.edu.ar/Carreras/Especializaciones/Redes_y_Seguridad/Trabajos_Finales/Pousa_Adrian.pdf).
30. (Pressman, 2005) **Pressman, Roger S. 2005.** *Ingeniería del software: Un enfoque práctico*. Sexta Edición. 2005. pág. 191 . ISBN: 9701054733.
31. (rae, 2005) **rae. 2005.** www.rae.es. [En línea] 10 de 2005. [Citado el: 20 de 05 de 2013.] <http://www.rae.es/dpd/>
32. (Ramírez, 2011) **Ramírez, Iván. 2011.** Softonic.com . [En línea] 9 de 08 de 2011. [Citado el: 20 de 02 de 2013.] <http://netbeans-ide.softonic.com/>.
33. (Rios, 2010) **Rios, Viviam Estrada. 2010.** *Sistema de Información empresarial ADMI 2013, Definición de conceptos*. Facultad de Ciencias Económicas y Administrativas, Universidad Austral de Chile. Chile : s.n., 2010.
34. (Rivas, y otros, 2008 pág. 4) **Rivas, Susana Marta y Gaitán, María Mercedes. 2008.** Universidad 2008. *6to Congreso Internacional de educación superior*. [En línea] 11-15 de febrero de 2008. [Citado el: 20 de 11 de 2012.]  
<http://revistas.mes.edu.cu/greenstone/collect/repo/import/repo/20100218/978959161165901149.pdf>.
35. (Rivera, 2010) **Rivera, Javier Fernández. 2010.** Bases de datos. *Modelo de datos*. [En línea] 2010. [Citado el: 20 de 05 de 2013.]  
<http://aurea.es/wp-content/uploads/modelodedatos.pdf>.
36. (Rosetta Stone, 2010) **Rosetta Stone. 2010.** rosettastone.com. *rosettastone*. [En línea] 2010. [Citado el: 10 de junio de 2013.]  
<http://www.rosettastone.com/lp/ingles/>.
37. (Segura, 2010) **Segura, Benjamín. 2010.** gratis.portalprogramas.com. [En línea] 25 de 10 de 2010. [Citado el: 20 de 02 de 2013.]  
<http://gratis.portalprogramas.com/abc-spelling.html>.
38. (Sommerville, 2005) **Sommerville, Ian. 2005.** *Ingeniería de software*. Madrid : s.n., 2005.  
[http://eva.uci.cu/file.php/161/Documentos/Materiales\\_basicos/Materiales\\_basicos\\_de\\_la\\_Unidad\\_2/Sommerville/Sommerville\\_Parte\\_I\\_Vision\\_General.pdf](http://eva.uci.cu/file.php/161/Documentos/Materiales_basicos/Materiales_basicos_de_la_Unidad_2/Sommerville/Sommerville_Parte_I_Vision_General.pdf). ISBN: 84-7829-074-5.

39. (Sparxsystems, 2007) —. **2007**. sparxsystems.com.ar. [En línea] 2007. [Citado el: 01 de 04 de 2013.] [http://www.sparxsystems.com.ar/resources/tutorial/use\\_case\\_model.html](http://www.sparxsystems.com.ar/resources/tutorial/use_case_model.html) .
40. (Sparxsystems, 2007a) —. **2007a**. Sparxsystems.com.ar. [En línea] 2007a. [Citado el: 01 de 05 de 2013.] <http://www.sparxsystems.com.ar/download/ayuda/index.html?deploymentdiagram.htm>.
41. (Sparxsystems, 2007b) —. **2007c**. sparxsystems.com.ar. [En línea] 2007c. [Citado el: 01 de 05 de 2013.] [http://www.sparxsystems.com.ar/resources/tutorial/uml2\\_componentdiagram.html](http://www.sparxsystems.com.ar/resources/tutorial/uml2_componentdiagram.html).
42. (Sparxsystems, 2008) **Sparxsystems. 2008**. sparxsystems. [En línea] 21 de 11 de 2008. [Citado el: 8 de 02 de 2013.] <http://www.sparxsystems.es/New/products/ea.html>.
43. (Tello, 2008) **Tello, Jesús Cáceres. 2008**. Universidad de Alcalá. *Diagrama de Casos de Uso*. [En línea] 2008. [Citado el: 10 de 05 de 2013.] <http://www2.uah.es/jcaceres/capsulas/DiagramaCasosDeUso.pdf>.
44. (UML, 2006) **Diagramas de Interacción. 2006**. UML y PD. *Tema 6: Diagramas de Interacción*. [En línea] 2006. [Citado el: 20 de 04 de 2013.] <http://www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r43251.PDF>.
45. (XML, 2010) **XML. 2010**. O'REILLY. *xml.com*. [En línea] 2010. [Citado el: 10 de 02 de 2013.] <http://www.xml.com/pub/a/98/10/guide0.html?page=2#AEN58>.
46. (Yanette, 2012 pág. 2) *Patrón Modelo Vista Controlador*. **Yanette Díaz González, Yenisleidy Fernández Romero. 2012**. 1, La Habana : s.n., 2012, Revista Digital de las Tecnologías de la Información y las Comunicaciones, Vol. 11, pág. 2. [revistatelematica.cujae.edu.cu/index.php/tele/article/download/15/10](http://revistatelematica.cujae.edu.cu/index.php/tele/article/download/15/10). ISSN: 1729-3804.

## Bibliografía

1. **Abrizaca, Lic. Eliza. 2009.** virtual.usalesiana.edu.bo. [En línea] 2009. [Citado el: 10 de 05 de 2013.] [virtual.usalesiana.edu.bo/web/practica/archiv/clases\\_2.doc?](http://virtual.usalesiana.edu.bo/web/practica/archiv/clases_2.doc?).
2. **Antonio Aliaga Ibarra, Marcos Agustín Miani Flores. 2008.** iessanvicente.com. [En línea] 21 de 01 de 2008. [Citado el: 10 de 05 de 2013.] <https://iessanvicente.com/colaboraciones/postgreSQL.pdf> .
3. **Arrocha, Juan Rafael Curbelo. 2008.** neleste.com. [En línea] 22 de 07 de 2008. [Citado el: 20 de 02 de 2013.] <http://www.neleste.com/modelo-vista-controlador/>.
4. **Bellizzi, Eric Ablett Dominick. 2011.** alfinsenior.wikispaces.com. [En línea] 25 de 11 de 2011. [Citado el: 20 de 11 de 2012.] <http://alfinsenior.wikispaces.com/Impacto+de+las+TIC+en+la+sociedad+%281%29...>
5. **Braña, Francisco. 2010.** gratis.portalprogramas.com. [En línea] 21 de 10 de 2010. [Citado el: 27 de 02 de 2013.] <http://gratis.portalprogramas.com/ABA-English-Course.html> .
6. **Brown. 2007.** ec.europa.eu. [En línea] 2007. [Citado el: 16 de 11 de 2012.] [http://ec.europa.eu/languages/documents/publications/learn\\_es.pdf](http://ec.europa.eu/languages/documents/publications/learn_es.pdf).
7. **Carretero, Valentín y Marín, Omar. 2008.** [En línea] 2008. [Citado el: 07 de 06 de 2013.] [http://iessanvicente.com/colaboraciones/berkeley\\_interbase.pdf](http://iessanvicente.com/colaboraciones/berkeley_interbase.pdf)..
8. **Ciclo de vida RUP . 2011 .** jummp.wordpress.com. [En línea] 6 de 04 de 2011 . [Citado el: 8 de 02 de 2013.] <http://jummp.wordpress.com/2011/04/06/desarrollo-de-software-ciclo-de-vida-rup-rational-unified-process/> .
9. **Craig, Larman. 1999.** *UML y Patrones*. s.l. : Segunda Edición ed. Prentice Hall, 1999.
10. **Cripto, Jimmy. 2010.** intercambiosvirtuales.org. [En línea] 2010. [Citado el: 20 de 02 de 2013.] <http://www.intercambiosvirtuales.org/software/microsoft-visual-studio-2010-ultimate-espanol>.
11. **Cubaeduca. 2009.** cubaeduca.cu. *Portal Educativo Cubano*. [En línea] 12 de 11 de 2009. [Citado el: 28 de 02 de 2013.] [http://www.cubaeduca.cu/index.php?option=com\\_content&view=article&id=4690&Itemid=153](http://www.cubaeduca.cu/index.php?option=com_content&view=article&id=4690&Itemid=153).
12. **Domínguez, Lic. Joel Fuentes. 2010.** pedagogiaprofesional.rimed.cu. [En línea] 2010. [Citado el: 20 de 11 de 2012.] <http://www.pedagogiaprofesional.rimed.cu/especial%202/Joe.htm>.
13. **Dwyer, Owen. 2010.** educacion-con-tic.blogspot.com. [En línea] 2010. [Citado el: 20 de 12 de 2012.] <http://educacion-con-tic.blogspot.com/2013/01/juegos-para-aprender-ingles.html>.

14. **Echeverr, Natalia. 2011.** slideshare.net. [En línea] 06 de 05 de 2011. [Citado el: 20 de 02 de 2013.] <http://www.slideshare.net/perezinho/lenguaje-de-programacion-c-7869638> .
15. **Eduticsantafe. 2012.** eduticsantafe.blogspot.com. [En línea] 2012. [Citado el: 9 de 02 de 2013.] <http://eduticsantafe.blogspot.com/> .
16. *Entornos Virtuales de Aprendizaje: Herramienta fundamental para desarrollar con mayor efectividad los procesos.* **León, Yordanis Medina y Ramírez, Yaneisy Castilla. 2010.** 1, La Habana : s.n., 2010, Serie Científica, Universidad de las Ciencias Informáticas, Vol. 3, pág. 12. <http://publicaciones.uci.cu/index.php/SC/issue/view/28>. ISSN: 2306-2495.
17. **Equipo de Softonic . 2011 .** sqlitemanager.softonic.com. [En línea] 05 de 08 de 2011 . [Citado el: 20 de 04 de 2013.] <http://sqlitemanager.softonic.com/> .
18. *Evaluación del desempeño para entorno virtual de autoaprendizaje de Inglés en la Universidad de* **Márquez, Yoan Martínez y Batista, Yalice Gámez. 2007.** 1, 2007, Serie Científica, Universidad de las Ciencias Informáticas, Vol. 1. ISSN: 2306-2495.
19. **Farràs, Pau. 2012.** blog.pequejuegos.com. [En línea] 29 de 05 de 2012. [Citado el: 20 de 11 de 2012.] <http://blog.pequejuegos.com/propuestas-para-mejorar-educacion-3-ingles-en-television-infantil/> .
20. **Ferguson, Jeff, Patterson, Brian y Beres, Jason. 2003.** *La biblia de C#.* [ed.] 15.28027 Juan Ignacio Luca de Tena. Madrid : EDICIONES ANAYA MULTIMEDIA (GRUPO ANAYA, S.A.), 2003. pág. 37. <http://bibliodoc.uci.cu/pdf/8441514844.pdf>.
21. **Ferrera, Lic. Miriam Cecilia Gómez. 2010.** pedagogiaprofesional.rimed.cu. [En línea] 2010. [Citado el: 16 de 11 de 2012.] <http://www.pedagogiaprofesional.rimed.cu/especial%202/Joe.htm>.
22. **Figuroa, Roberth G., Solís, Camilo J y Cabrera, Armando A. 2011.** *Metodologías tradicionales VS. Metodologías ágiles.* 2011. pág. 2. [http://eva.uci.cu/file.php/161/Documentos/Materiales\\_complementarios/UD\\_1\\_Procesos/Metodologias/METODOLOGIAS\\_TRADICIONALES\\_VS.\\_METODOLOGIAS\\_AGILES.pdf](http://eva.uci.cu/file.php/161/Documentos/Materiales_complementarios/UD_1_Procesos/Metodologias/METODOLOGIAS_TRADICIONALES_VS._METODOLOGIAS_AGILES.pdf).
23. **Filip, David. 2010.** Imendi.com. [En línea] 2010. [Citado el: 20 de 02 de 2013.] [imendi.com/selectlengs](http://imendi.com/selectlengs) .
24. **Fire, Angel. 2011.** geektheplanet.net. [En línea] 11 de 05 de 2011. [Citado el: 25 de 05 de 2013.] <http://geektheplanet.net/5462/patrones-gof.xhtml>.

25. **García, Olga. 2010.** gestiopolis.com. [En línea] 2010. [Citado el: 20 de 11 de 2012.] <http://www.gestiopolis.com/administracion-estrategia-2/impacto-de-las-tic-en-el-mundo-de-hoy.pdf> .
26. **Globedia. 2009.** es.globedia.com. [En línea] 7 de 05 de 2009. [Citado el: 21 de 11 de 2012.] <http://es.globedia.com/el-ingles-en-la-educacion>.
27. **Gómez, José Ramón. 2007.** boj.pntic.mec.es. [En línea] 2 de 03 de 2007. [Citado el: 20 de 11 de 201.] <http://boj.pntic.mec.es/jgomez46/ticedu.htm#Uso>.
28. **González, YD. 2012.** revistatelematica.cujae.edu.cu. [En línea] 2012. [Citado el: 20 de 05 de 2013.] [revistatelematica.cujae.edu.cu/index.php/tele/article/download/15/10](http://revistatelematica.cujae.edu.cu/index.php/tele/article/download/15/10).
29. **Graells, Dr. Pere Marquès. 2011.** [En línea] 7 de 08 de 2011. [Citado el: 9 de 01 de 2013.] <http://peremarques.pangea.org/siyedu.htm> .
30. **GSInnova. 2012.** rational.com.ar. [En línea] 2012. [Citado el: 20 de 02 de 2013.] <http://www.rational.com.ar/herramientas/rup.html>.
31. **Guerra, Lic. Jorge Guerra. 2007.** es.scribd.com. [En línea] 13 de 10 de 2007. [Citado el: 22 de 04 de 2013.] <http://es.scribd.com/doc/395783/RUP-etapa-diseno>.
32. **Guerrero, Rafael Martinez. 2009.** PostgreSQL-es. [En línea] 2009. [Citado el: 20 de 02 de 2013.] [http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql).
33. **Guia-ubuntu. 2011.** guia-ubuntu. [En línea] 29 de julio de 2011. [Citado el: 10 de 05 de 2013.] <http://www.guia-ubuntu.com/index.php?title=MonoDevelop>.
34. —. **2008.** guia-ubuntu. [En línea] 10 de 03 de 2008. [Citado el: 26 de 05 de 2013.] [http://www.guia-ubuntu.com/index.php?title=PgAdmin\\_III](http://www.guia-ubuntu.com/index.php?title=PgAdmin_III).
35. **Hidalgo, Daira Figueroa, Ortiz, Yurisbel Vega y Fernández, Vladimir Martell. 2010.** Propuesta de diseño para proyectos informáticos que utilizan Symfony como Framework. [En línea] 2010. [Citado el: 20 de 05 de 2013.] [publicaciones.uci.cu/index.php/SC/article/download/208/228](http://publicaciones.uci.cu/index.php/SC/article/download/208/228).
36. **Hipp, D. Ri-chard. 2011.** sqlite-latino.blogspot.com. [En línea] 12 de 04 de 2011. [Citado el: 05 de 03 de 2013.] <http://sqlite-latino.blogspot.com/2011/04/sqlite-articulo-en-pdf.html>.
37. **Ilustrados. 2009.** ilustrados.com. [En línea] 2009. [Citado el: 9 de 01 de 2013.] <http://www.ilustrados.com/tema/10871/motivacion-aprendizaje-idioma-extranjero-Universidad-Ciencias.html> .
38. **Informáticas, Universidad de las Ciencias.** Entorno Virtual de Aprendizaje. EVA. [En línea] UCI. [Citado el: 19 de 02 de 2013.] <http://eva.uci.cu>.



39. **InstEd. 2009** . blogs.rimed.cu. [En línea] 12 de 11 de 2009 . [Citado el: 28 de 02 de 2013.] <http://blogs.rimed.cu/futuro/files/2009/11/Manual-Sunrise.pdf> .
40. **Itescam. 2006**. itescam.edu.mx. [En línea] 2006. [Citado el: 20 de 04 de 2013.] <http://www.itecam.edu.mx/principal/sylabus/fpdb/recursos/r43251.PDF>.
41. **Jacobson, James. 2000**. *El proceso unificado de desarrollo de software*. Madrid : Pearson Educación, S.A, 2000.
42. —. **2004**. *El proceso unificado de desarrollo de software*. [ed.] Addison Wesley. 2004.
43. **Java. 2009**. lenguajes-de-programacion.com. *Lenguajes de programación*. [En línea] 2009. [Citado el: 20 de 11 de 2013.] <http://www.lenguajes-de-programacion.com/programacion-java.shtml>.
44. *La Tecnología informática en la sociedad cubana*. **Ferrera, Miriam Cecilia Gómez y Domínguez, Joel Fuentes. 2010**. [ed.] Juana Ivis Imamura Díaz. Ciudad de la Habana : s.n., 2010, Pedagogía Profesional, Vols. Edición especial,2. <http://www.pedagogiaprofesional.rimed.cu/especial%202/Joe.htm>. ISSN 1684-5765.
45. **Labbe, Matias. 2011**. ahoraentiendoingles.com. [En línea] 28 de 08 de 2011. [Citado el: 18 de 11 de 2012.] [http://www.ahoraentiendoingles.com/pages\\_other/not-a-member.html](http://www.ahoraentiendoingles.com/pages_other/not-a-member.html) .
46. **Las TIC en educacion. 2007**. boj.pntic.mec.es. [En línea] 2 de 03 de 2007. [Citado el: 18 de 11 de 2012.] <http://boj.pntic.mec.es/jgomez46/ticedu.htm>.
47. **mansioningles. 2011**. mansioningles.com. [En línea] 2011. [Citado el: 18 de 11 de 2012.] <http://www.mansioningles.com/index.htm>.
48. **Martínez, Prof. Ivette Carolina. 2009**. Clase 6: Modelo conceptual/Modelo de dominio. [En línea] 2009. [Citado el: 10 de 03 de 2013.] <http://www.itecam.edu.mx/principal/sylabus/fpdb/recursos/r88846.PDF>.
49. —. **2009**. itescam.edu.mx. [En línea] 2009. [Citado el: 10 de 03 de 2013.] <http://www.itecam.edu.mx/principal/sylabus/fpdb/recursos/r88846.PDF>.
50. **Masadelante. 2010**. masadelante.com. [En línea] 2010. [Citado el: 20 de 02 de 2013.] <http://www.masadelante.com/faqs/html> .
51. **Mastermagazine. 2007**. mastermagazine.info. [En línea] 2007. [Citado el: 10 de 02 de 2013.] <http://www.mastermagazine.info/termino/7006.php>.
52. **Mayo, Joe. 2011**. csharp-station.com. [En línea] 9 de 02 de 2011. [Citado el: 14 de 02 de 2013.] [csharp-station.com/Tutorial/CSharp/Lesson11](http://csharp-station.com/Tutorial/CSharp/Lesson11).

53. **Memrise. 2010.** memrise.com. [En línea] 2010. [Citado el: 20 de 02 de 2013.] [www.memrise.com/welcome/](http://www.memrise.com/welcome/) .
54. **Michae. 2012.** computer-programming-forum.com. [En línea] 23 de 08 de 2012. [Citado el: 14 de 02 de 2013.] [computer-programming-forum.com/.../e64fa529..](http://computer-programming-forum.com/.../e64fa529..)
55. **MongoDB. 2010.** mongoDB. [En línea] 2010. [Citado el: 07 de 06 de 2013.] <http://www.mongodb.org>.
56. **Monodevelop. 2010.** monodevelop.com. [En línea] 6 de 12 de 2010. [Citado el: 20 de 2 de 2013.] <http://monodevelop.com/> .
57. **Montiel, Daniel Ponsoda. 2008.** Introducción a SQLite. [En línea] 23 de enero de 2008. [Citado el: 06 de 06 de 2013.] <https://iessanvicente.com/colaboraciones/sqlite>.
58. **Mylingo. 2011.** mylingo.org. [En línea] 2011. [Citado el: 20 de 02 de 2013.] [www.mylingo.org/es](http://www.mylingo.org/es) .
59. **Nuryr. 2008.** tic\_nury.zoomblog.com. [En línea] 31 de 10 de 2008. [Citado el: 20 de 11 de 2012.] [http://tic\\_nury.zoomblog.com/archivo/2008/10/31/importancia-de-las-Tics-para-el-Desarr.html](http://tic_nury.zoomblog.com/archivo/2008/10/31/importancia-de-las-Tics-para-el-Desarr.html).
60. **Olson, Philip. 2013.** php.net . [En línea] 22 de 02 de 2013. [Citado el: 27 de 02 de 2013.] <http://www.php.net/manual/es/intro-what-is.php> .
61. **Orallo, Enrique Hernández. 2007.** *El Lenguaje Unificado de Modelado(UML)*. España : Valencia s.n, 2007.
62. *Patrón Modelo Vista Controlador. Yanette Díaz González, Yenisleidy Fernández Romero. 2012.* 1, La Habana : s.n., 2012, Revista Digital de las Tecnologías de la Información y las Comunicaciones, Vol. 11, pág. 2. [revistatelematica.cujae.edu.cu/index.php/tele/article/download/15/10](http://revistatelematica.cujae.edu.cu/index.php/tele/article/download/15/10). ISSN: 1729-3804.
63. **Postgresql. 2010.** postgresql. [En línea] 10 de 10 de 2010. [Citado el: 21 de 01 de 2013.] <http://www.postgresql.org/about/>.
64. **Pousa, Lic. Adrián. 2011.** ALGORITMO DE CIFRADO SIMÉTRICO AES. *ACELERACIÓN DE TIEMPO DE CÓMPUTO SOBRE ARQUITECTURAS MULTICORE*. [En línea] diciembre de 2011. [Citado el: 20 de 05 de 2013.] [http://postgrado.info.unlp.edu.ar/Carreras/Especializaciones/Redes\\_y\\_Seguridad/Trabajos\\_Finales/Pousa\\_Adrian.pdf](http://postgrado.info.unlp.edu.ar/Carreras/Especializaciones/Redes_y_Seguridad/Trabajos_Finales/Pousa_Adrian.pdf).
65. **Pressman, Roger S. 2005.** *Ingeniería del software: Un enfoque práctico*. Sexta Edición. 2005. pág. 191 . ISBN: 9701054733.



66. **Quirón. 2009.** epidataconsulting.com. [En línea] 4 de 10 de 2009. [Citado el: 10 de 02 de 2013.] [http://www.epidataconsulting.com/tikiwiki/tiki-read\\_article.php?articleId=15](http://www.epidataconsulting.com/tikiwiki/tiki-read_article.php?articleId=15).
67. **rae. 2005.** www.rae.es. [En línea] 10 de 2005. [Citado el: 20 de 05 de 2013.] <http://www.rae.es/dpd/>.
68. **Rafaelma. 2010 .** postgresql.org.es. [En línea] 02 de 10 de 2010 . [Citado el: 10 de 05 de 2013.] [http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql) .
69. **Ramírez, Iván. 2011.** Softonic.com . [En línea] 9 de 08 de 2011. [Citado el: 20 de 02 de 2013.] <http://netbeans-ide.softonic.com/>.
70. **Ramos, Aurora Carretero. 2007.** quadernsdigitals.net. [En línea] 2007. [Citado el: 20 de 11 de 2012.] [http://quadernsdigitals.net/datos\\_web/hemeroteca/r\\_59/nr\\_642/a\\_8656/8656.html](http://quadernsdigitals.net/datos_web/hemeroteca/r_59/nr_642/a_8656/8656.html).
71. **Rios, Viviam Estrada. 2010.** *Sistema de Información empresarial ADMI 2013, Definición de conceptos.* Facultad de Ciencias Económicas y Administrativas, Universidad Austral de Chile. Chile : s.n., 2010.
72. **Rivas, Susana Marta y Gaitán, María Mercedes. 2008.** Universidad 2008. *6to Congreso Internacional de educación superior.* [En línea] 11-15 de febrero de 2008. [Citado el: 20 de 11 de 2012.] <http://revistas.mes.edu.cu/greenstone/collect/repo/import/repo/20100218/978959161165901149.pdf> .
73. **Rivera, Javier Fernández. 2010.** Bases de datos. *Modelo de datos.* [En línea] 2010. [Citado el: 20 de 05 de 2013.] <http://aurea.es/wp-content/uploads/modelodedatos.pdf>..
74. **Rodríguez, Alberto. 1997.** Diseño de Aplicaciones Three Tier. *Arquitectura Three-Tier.* [En línea] FoxPress, septiembre de 1997. [Citado el: 14 de junio de 2013.] <http://www.fpress.com/revista/Num9711/Nov97.htm>.
75. **Rodríguez, Yhon Fredy Medina. 2012.** msdn.microsoft.com. [En línea] 04 de 10 de 2012. [Citado el: 20 de 02 de 2013.] <http://msdn.microsoft.com/es-es/library/vstudio/6x6bk1f4%28v=vs.100%29.aspx>.
76. **Rojas, Carolina. 2008.** slideshare.net. [En línea] 18 de 06 de 2008. [Citado el: 20 de 04 de 2013.] <http://www.slideshare.net/chiki.carito/modelado-del-analisis> .
77. **Rosetta Stone. 2010.** rosettastone.com. *rosettastone.* [En línea] 2010. [Citado el: 10 de junio de 2013.] <http://www.rosettastone.com/lp/ingles/>.

78. **Sebastián, Juan. 2010.** Comusoft.com. [En línea] 13 de 11 de 2010. [Citado el: 20 de 02 de 2013.] <http://www.comusoft.com/modelo-vista-controlador-definicion-y-caracteristicas>.
79. **Segura, Benjamín. 2010.** gratis.portalprogramas.com. [En línea] 25 de 10 de 2010. [Citado el: 20 de 02 de 2013.] <http://gratis.portalprogramas.com/abc-spelling.html>.
80. **Sommerville, Ian. 2005.** *Ingeniería de software*. Madrid : s.n., 2005. [http://eva.uci.cu/file.php/161/Documentos/Materiales\\_basicos/Materiales\\_basicos\\_de\\_la\\_Unidad\\_2/Sommerville/Sommerville\\_Parte\\_I\\_Vision\\_General.pdf](http://eva.uci.cu/file.php/161/Documentos/Materiales_basicos/Materiales_basicos_de_la_Unidad_2/Sommerville/Sommerville_Parte_I_Vision_General.pdf). ISBN: 84-7829-074-5.
81. —. **2004.** lsi.ugr.es. [En línea] 2004. [Citado el: 01 de 04 de 2013.] <http://lsi.ugr.es/~ig1/docis/requeintro.pdf> .
82. **Sparxsystems. 2008.** sparxsystems. [En línea] 21 de 11 de 2008. [Citado el: 8 de 02 de 2013.] <http://www.sparxsystems.es/New/products/ea.html>.
83. —. **2007.** sparxsystems.com.ar. [En línea] 2007. [Citado el: 01 de 04 de 2013.] [http://www.sparxsystems.com.ar/resources/tutorial/use\\_case\\_model.html](http://www.sparxsystems.com.ar/resources/tutorial/use_case_model.html) .
84. —. **2007b.** sparxsystems.com.ar. [En línea] 2007b. [Citado el: 01 de 05 de 2013.] [http://www.sparxsystems.com.ar/resources/tutorial/uml2\\_componentdiagram.html](http://www.sparxsystems.com.ar/resources/tutorial/uml2_componentdiagram.html).
85. —. **2007a.** Sparxsystems.com.ar. [En línea] 2007a. [Citado el: 01 de 05 de 2013.] <http://www.sparxsystems.com.ar/download/ayuda/index.html?deploymentdiagram.htm>.
86. **Steele, Oliver. 2009.** leandroirarte.com.ar. [En línea] 2009. [Citado el: 01 de 05 de 2013.] [http://www.leandroirarte.com.ar/spanish/web\\_mvc.php#1\\_auth](http://www.leandroirarte.com.ar/spanish/web_mvc.php#1_auth) .
87. **Stoner, James. 1990.** *Administracion (Primera Parte)*. Habana : Alejo Carpentier, 1990.
88. **Tello, Jesús Cáceres. 2008.** Universidad de Alcalá. *Diagrama de Casos de Uso*. [En línea] 2008. [Citado el: 10 de 05 de 2013.] <http://www2.uah.es/jcaceres/capsulas/DiagramaCasosDeUso.pdf>.
89. **Tutorial de UML. 2010.** users.dcc.uchile.c. [En línea] 2010. [Citado el: 10 de 02 de 2013.] <http://users.dcc.uchile.cl/~psalinas/uml/introduccion.html>.
90. **UML. 2006.** UML y PD. *Tema 6: Diagramas de Interacción*. [En línea] 2006. [Citado el: 20 de 04 de 2013.] <http://www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r43251.PDF>.
91. **Violeta. 2008** . scribd.com. [En línea] 20 de 10 de 2008 . [Citado el: 20 de 02 de 2013.] <http://es.scribd.com/doc/7411856/Caracteristicas-de-C>.
92. **Visual Paradigm. 2008.** visual-paradigm.com. [En línea] 2008. [Citado el: 20 de 02 de 2013.] <http://www.visual-paradigm.com/product/vpuml/> .

93. **XML. 2010.** O'REILLY. *xml.com*. [En línea] 2010. [Citado el: 10 de 02 de 2013.]  
<http://www.xml.com/pub/a/98/10/guide0.html?page=2#AEN58>.

## Anexos

### Anexo 1. Encuesta

Encuesta para conocer los problemas existentes sobre el aprendizaje del idioma Inglés desde la producción  
Departamento Productivo: Sistema de Apoyo a la Salud.

Rol: \_\_\_\_\_

A continuación se proponen un conjunto de preguntas para conocer el grado de satisfacción de los estudiantes por el estudio del Inglés de temas productivos.

1. \*¿Te gusta utilizar tu tiempo de la producción para estudiar herramientas, tecnologías, lenguajes de programación en idioma Inglés? Marque con una x la respuesta correspondiente.

\_\_\_\_\_ Si          \_\_\_\_\_ No \_\_\_\_\_ No sé

2. ¿Cómo cree que le fuera más fácil el estudio de temas productivos en el idioma Inglés, con una aplicación Desktop o Web? Marque con una x la respuesta correspondiente.

\_\_\_\_\_ Desktop          \_\_\_\_\_ Web          ¿Por qué?

3. ¿Conoces algunas herramientas utilizada en la universidad que permita el estudio de temas productivos en el idioma Inglés, tales como: herramientas, tecnologías, lenguajes de programación y negocio de los sistemas existentes desarrollados en la Universidad, así como del negocio en el que se fundamentan? Marque con una x la respuesta correspondiente.

\_\_\_\_\_ Si          \_\_\_\_\_ No \_\_\_\_\_ No sé

a) Si la respuesta es afirmativa liste los que conozca.

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

4. ¿Cuáles son los principales problemas que usted presenta con el Inglés en los proyectos productivos? Marque con una x la respuesta correspondiente.

\_\_\_\_\_ No se traducir.

\_\_\_\_\_ No tengo conocimiento de donde buscar.

\_\_\_\_\_ Presento mucha dependencia de traductores.

\_\_\_\_\_ Tengo mala preparación a la hora de comunicarme en Inglés

\_\_\_\_\_ Tengo que buscar bibliográfica en español a la hora de estudiar las tecnologías con las que trabajo en el departamento.

Otros: \_\_\_\_\_

5. ¿Cuáles son los temas que quisiera estudiar en idioma Inglés para producir mejor? Marque con una x la respuesta correspondiente.

- Tecnologías
- Herramientas
- Lenguajes de programación
- Negocio del sistema que se implementa en los proyectos productivos

Otros: \_\_\_\_\_

6. \*Si contara con una aplicación portable con ejercicios en Inglés, de temas de la producción para su estudio independiente y profesor que lo ayudara a mejorar el Inglés. ¿Usted estudiaría? Marque con una x la respuesta correspondiente.

Si  No  No sé

7. ¿En qué lugar de la UCI le gusta estudiar Inglés? Marque con una x la respuesta correspondiente.

- Apartamento
- Laboratorio de proyecto
- Laboratorio docente

Otros: \_\_\_\_\_

8. ¿Qué tipo de actividades le gustaría que tuviera una aplicación que le permitiera estudiar temas de la producción en Inglés? Marque con una x la respuesta correspondiente.

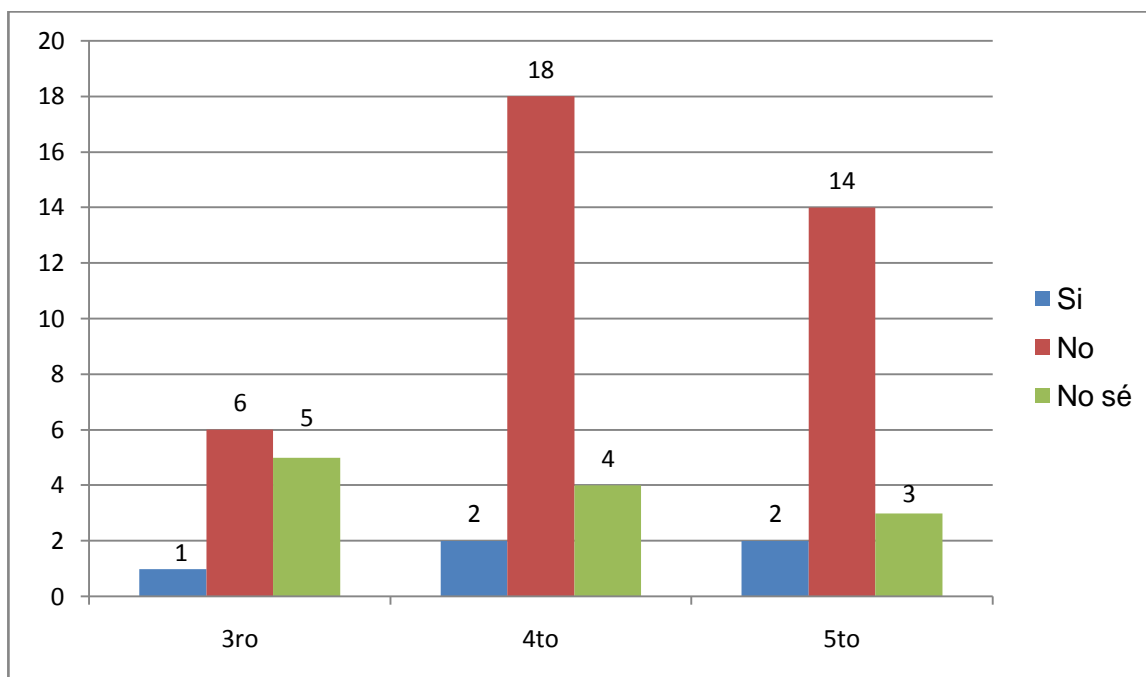
- Marcar
- Enlazar
- Verdadero o Falso
- Redactar
- Completar

Otras: \_\_\_\_\_

9. \*¿Te gusta estudiar Inglés? específicamente temas que tengan que ver con la producción. Marque con una x la respuesta correspondiente.

- Me gusta mucho
- No me gusta tanto
- Me da lo mismo
- Me disgusta más de lo que me gusta
- No me gusta nada
- No sé

**Anexo 2. Porciento de conocimiento acerca de herramientas utilizadas en la UCI para el estudio del Inglés desde la producción**



**Anexo 3. Técnica de V.A. ladov**

Cuadro lógico de V.A. ladov (con tres preguntas ocultas de la encuesta)

|  |   |       |    |       |       |    |    |       |    |
|--|---|-------|----|-------|-------|----|----|-------|----|
|  | 3. ¿Te gusta utilizar tu tiempo de la producción para estudiar herramientas, tecnologías, lenguajes de programación en idioma Inglés?   |       |    |       |       |    |    |       |    |
|  | No  |       |    | No sé |       |    | Si |       |    |
| 9. ¿Te gusta estudiar Inglés?<br>específicamente temas que tengan que ver con la producción. | 6. Si contara con una aplicación portable con ejercicios en Inglés, de temas de la producción para su estudio independiente y profesor que lo ayudara a mejorar el Inglés. ¿Usted estudiaría? |       |    |       |       |    |    |       |    |
|  | Si  | No sé | No | Si    | No sé | No | Si | No sé | No |
| Me gusta mucho   | 1   | 2     | 6  | 2     | 2     | 6  | 6  | 6     | 6  |

|                                    |   |   |   |   |   |   |   |   |   |
|------------------------------------|---|---|---|---|---|---|---|---|---|
| No me gusta tanto                  | 2 | 2 | 3 | 2 | 3 | 3 | 6 | 3 | 6 |
| Me da lo mismo                     | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| Me disgusta más de lo que me gusta | 6 | 3 | 6 | 3 | 4 | 4 | 3 | 4 | 4 |
| No me gusta nada                   | 6 | 6 | 6 | 6 | 4 | 4 | 6 | 4 | 5 |
| No sé                              | 2 | 3 | 6 | 3 | 3 | 3 | 6 | 3 | 4 |

El número resultante de la interrelación de las tres preguntas indica la posición de cada sujeto en la escala de satisfacción.

La escala de satisfacción es la siguiente:

1. Clara satisfacción
2. Más satisfecho que insatisfecho
3. No definida
4. Más insatisfecho que satisfecho
5. Clara insatisfacción
6. Contradictoria

➤ Tabla para obtener el índice de satisfacción grupal:

|      |                                 |
|------|---------------------------------|
| +1   | Máximo de satisfacción          |
| 0,5  | Más satisfecho que insatisfecho |
| 0    | No definido y contradictorio    |
| -0,5 | Más insatisfecho que satisfecho |
| -1   | Máxima insatisfacción           |

➤ Fórmula para calcular el índice de satisfacción grupal:

A = cantidad de estudiantes en la primera posición de la escala (3)

B = cantidad de estudiantes en la segunda posición de la escala (8)

C = cantidad de estudiantes en la tercera o sexta posición de la escala (38)

D = cantidad de estudiantes en la cuarta posición de la escala (0)

E = cantidad de estudiantes en la quinta posición de la escala (3)

N = total de estudiantes (52)

$$\text{ISG} = \frac{3 (+1) + 8 (+0.5) + 38 (0) + 0 (-0.5) + 3 (-1)}{52}$$