



**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**  
**FACULTAD 1**

***Análisis y Diseño de los módulos tutor y dominio de un Sistema Tutorial Inteligente para la enseñanza de la Inteligencia Artificial 1.***

Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas

**Autor:** Olga João Eduardo

**Tutores:**

MsC. Maidelis Milanés Luque

Ing. Estela Odelsa Martín Coronel

La Habana, junio de 2020.

“Año 62 de la Revolución”



### Declaración jurada de autoría

Declaro por este medio, que yo Olga João Eduardo, soy la autora principal del trabajo de diploma titulado **“Análisis y Diseño de los módulos tutor y dominio de un Sistema Tutorial Inteligente para la enseñanza de la Inteligencia Artificial 1”**, y autorizo a la Universidad de las Ciencias Informáticas a hacer uso de la misma en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Autor

---

Olga João Eduardo

Tutor

---

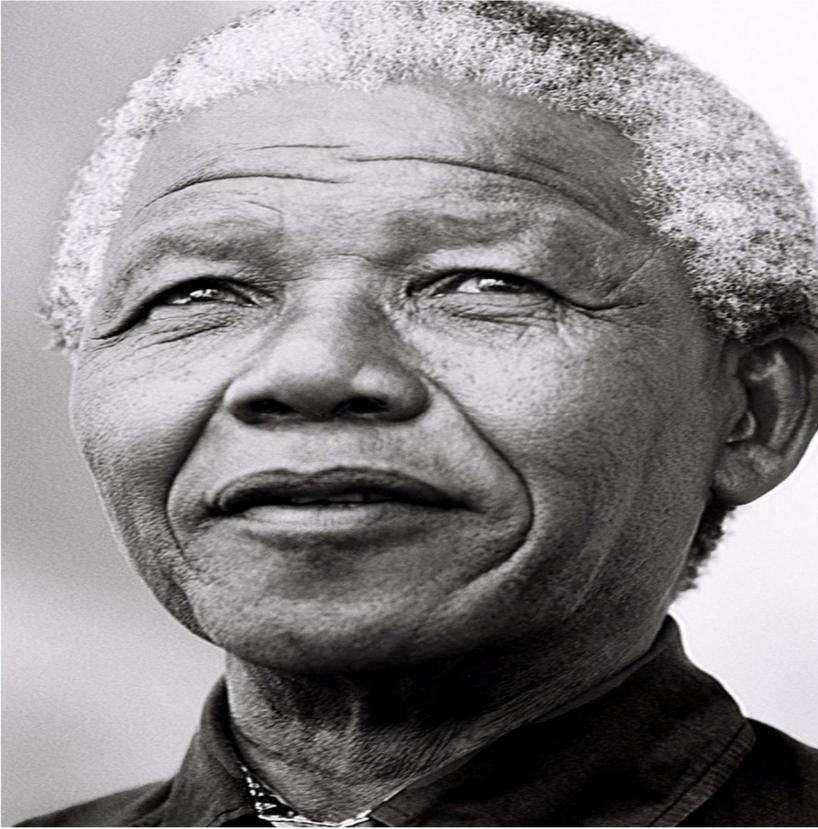
MSc. Maidelis Milanés Luque

Tutor

---

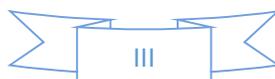
Ing. Estela Odelsa Martín Coronel





*“La educación y la enseñanza son las armas más poderosas que puedes usar para cambiar el mundo”.*  
*“Todo parece imposible hasta que sucede”.*

***Nelson Mandela (1918-2013)***



## ***Agradecimientos***

*Primeramente agradecer a Dios por haberme dado salud, fuerza, voluntad y coraje para poder lograr mis objetivos y metas a través de este arduo camino hacia aquí.*

*A mis padres Rebeca y Ricardo mil gracias por todo el amor y el apoyo incondicional que me han dedicado durante toda mi vida y todos estos años de la carrera aun estando lejos, por formarme y convertirme en la persona de bien que soy hoy.*

*A mis hermanas Juliana, Albertina y Felicidade por ser mi inspiración y un ejemplo a seguir.*

*A mis hermanos, a mi sobrino y a mi abuelita gracias por el apoyo.*

*A mis compañeros angolanos y a mis compañeros de clase mis amigos de toda la vida mil gracias por todo.*

*A mis compañeras de apartamento, Leumira, Debora, en especial a Ludmila por haber estado y llegado conmigo hasta esta etapa final y demostrar ser una gran amiga durante estos años, muchas gracias por el apoyo.*

*A mis tutoras Maidelis y Estela por estar pendiente de mi durante el proceso de desarrollo de la tesis muchas gracias.*

*A toda mi familia y amigos que de alguna forma me han dado su apoyo durante mis estudios.*

*A todos los profesores y colectivo de la facultad y de la universidad que han contribuido a mi formación como profesional en estos 5 años de estudios mil gracias.*

## **Dedicatoria**

*Dedico mi tesis a mis padres, las personas que más amo en el mundo por haberme dado la vida, por su amor y todo el apoyo durante toda mi vida, por confiar en mí y darme fuerza siempre para salir adelante.*



## Resumen

En los procesos de enseñanza-aprendizaje surgen diversos problemas en cuanto al entendimiento y la comprensión del conocimiento. Estas dificultades radican principalmente en que todas las personas tienen un estilo de aprendizaje diferente y los métodos clásicos de enseñanza no cubren sus necesidades particulares. El desarrollo de la tecnología ha impulsado la creación de herramientas que brindan una solución eficiente a dicha problemática: los Sistemas Tutores Inteligentes.

La presente investigación plantea como objetivo realizar el análisis y diseño de los módulos tutor y dominio de un Sistema Tutorial Inteligente como apoyo al proceso de enseñanza-aprendizaje de la asignatura Inteligencia Artificial<sup>1</sup>, en la Universidad de las Ciencias Informáticas, que ayude a consolidar los contenidos del plan de estudio definidos para la asignatura, utilizando para ello el Razonamiento Basado en Reglas como técnica de inteligencia artificial. Con la implementación futura de este sistema digital tutorial se pretende mejorar la relación entre el docente y el estudiante y facilitar la obtención de información académica del estudiante y lograr que este tenga una mejor experiencia educativa que lo formará como profesional altamente calificado. Se obtendrán como resultado todos los artefactos necesarios para la posterior implementación de los módulos tutor y dominio de un sistema tutorial inteligente para la enseñanza de la inteligencia artificial 1.

**Palabras clave:** Aprendizaje, Educación, Enseñanza, Inteligencia Artificial, Módulo dominio, Módulo tutor, Sistema Basado en Reglas, Sistemas Tutores Inteligentes.

## **Abstract**

In the teaching-learning processes, various problems arise in terms of understanding and understanding of knowledge. These difficulties mainly lie in the fact that all people have a different learning style and the classical teaching methods do not cover their particular needs. The development of technology has fostered the creation of tools that provide an efficient solution to this problem: Intelligent Mentoring Systems.

The objective of this research is to carry out the analysis and design of the tutor modules and mastery of an Intelligent Tutorial System to support the teaching-learning process of the Artificial Intelligence subject<sup>1</sup>, at the University of Computer Sciences, which helps to consolidate the contents of the study plan defined for the subject, using for it the Rule Based Reasoning as an artificial intelligence technique. With the future implementation of this digital tutorial system, the aim is to improve the relationship between the teacher and the student, facilitate obtaining the student's academic information, and ensure that the student has a better educational experience that will train him or her as a highly qualified professional. As a result, all the necessary artifacts will be obtained for the subsequent implementation of the tutor modules and mastery of an intelligent tutorial system for teaching artificial intelligence 1.

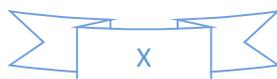
**Key words:** Learning, Education, Teaching, Artificial Intelligence, Tutor module, Domain module, Rule Based System, Intelligent Tutor Systems.

# ÍNDICE

|   |    |
|---|----|
| Introducción-----   | 1  |
| CAPÍTULO 1. Fundamentación teórica del análisis y diseño de los módulos tutor y dominio de un Sistema Tutorial Inteligente en el proceso de enseñanza - aprendizaje de la Inteligencia Artificial I ----- | 7  |
| 1.1- Introducción.-----   | 7  |
| 1.2- Sistemas Tutores Inteligentes.-----  | 7  |
| 1.3- Técnicas de Inteligencia Artificial usadas para el desarrollo de los módulos tutor y dominio de un STI. ----   | 13 |
| 1.3.1- Fundamentos de la selección-----   | 18 |
| 1.4- Análisis de sistemas similares-----  | 19 |
| 1.4.1- Análisis a nivel Internacional -----   | 19 |
| 1.4.2- Análisis a nivel nacional-----   | 21 |
| 1.4.3- A nivel UCI-----   | 22 |
| 1.4.4- Valoración de los sistemas similares -----   | 25 |
| 1.5- Metodologías de desarrollo de software y tecnologías a utilizar -----  | 25 |
| 1.5.1- Proceso Unificado Ágil (AUP)-----  | 26 |
| 1.5.2- Fases de la metodología-----   | 26 |
| 1.5.3- Características por escenarios -----   | 26 |
| 1.5.4- Fundamentos de la selección-----   | 27 |
| 1.6- Lenguajes de Programación-----   | 27 |
| 1.6.1- Python v2019-----  | 28 |
| 1.6.2- Fundamentos de la selección-----   | 28 |
| 1.6.3. Lenguaje marcado de hipertexto (HTML 5)-----   | 29 |
| 1.7- Entorno de desarrollo Integrado (IDE)-----   | 29 |
| 1.7.1- PyCharm v9.2.1. -----  | 29 |
| 1.8- Marco de Trabajo. -----  | 30 |
| 1.8.1- Django v1.8 -----  | 30 |
| 1.9- Lenguaje de Modelado -----   | 30 |
| 1.9.1. Herramienta para el modelado -----   | 31 |
| 1.9.2- Visual Paradigm v8.0 -----   | 31 |

|   |    |
|---|----|
| 1.10- Sistema Gestor de Base de Datos-----  | 31 |
| 1.10.1- MySQL v5.7-----   | 32 |
| 1.12- PhpMyAdmin v4.8.3-----  | 33 |
| 1.13- Conclusiones del capítulo-----  | 33 |
| Capítulo 2: Diseño y análisis de la propuesta de solución para el desarrollo de los módulos tutor y dominio de un sistema tutorial inteligente para la enseñanza de la inteligencia artificial I----- | 34 |
| 2.1- Introducción-----  | 34 |
| 2.2- Propuesta de Solución-----   | 34 |
| 2.2.1- Cuestionario Diagnóstico-----  | 35 |
| 2.2.2- Indicadores y Descriptores de la dimensión Acceso a la Información:-----   | 35 |
| 2.2.3- Medida de Valor para la calificación del diagnóstico:-----   | 36 |
| 2.3. Sistema Basado en reglas-----  | 36 |
| 2.4- Modelo Conceptual-----   | 37 |
| 2.5- Personas relacionadas con el sistema-----  | 39 |
| 2.6- Lista de Reserva de Productos.-----  | 40 |
| 2.6.1- Requisitos funcionales-----  | 41 |
| 2.6.2- Requisitos no funcionales-----   | 45 |
| 2.7- Casos de Uso del Sistema-----  | 51 |
| 2.8- Descripción de los casos de uso del sistema.-----  | 52 |
| 2.9- Diagrama de Clases-----  | 55 |
| 2.9.1- Diagrama de Clases de diseño-----  | 55 |
| 2.10- Modelo de datos-----  | 57 |
| 2.11- Modelo de Despliegue-----   | 58 |
| 2.12- Patrón Arquitectónico-----  | 59 |
| 2.13- Patrones de Diseño-----   | 60 |
| 2.13.1- Patrones GRASP-----   | 60 |
| 2.13.2- Patrones GOF-----   | 62 |
| 2.14- Diagrama de Componentes-----  | 62 |
| 2.15- Conclusiones del Capítulo-----  | 63 |
| Conclusiones generales-----   | 64 |

|   |    |
|---|----|
| Recomendaciones -----   | 65 |
| Referencias Bibliográficas-----   | 66 |
| Bibliografía-----   | 70 |
| Anexos-----   | 75 |
| Anexo 1: Preguntas de la entrevista realizada a los clientes, que en este caso son los profesores de IA:----- | 75 |
| Anexo 2: Casos de uso del Sistema: -----  | 76 |



## Índice de tablas

|   |    |
|---|----|
| Tabla 1: Estudio de los sistemas similares.....                       | 22 |
| Tabla 2: Personas relacionadas con el sistema. ....                   | 40 |
| Tabla 3: Descripción de los requisitos funcionales.....               | 42 |
| Tabla 4: Requisito no funcional de Software.....                      | 46 |
| Tabla 5: Requisito no funcional de Usabilidad (comprensión).....      | 47 |
| Tabla 6: Requisito no funcional de Usabilidad (Operabilidad).....     | 48 |
| Tabla 7: Requisito no funcional de Hardware.....                      | 49 |
| Tabla 8: Requisito no funcional de Portabilidad (adaptabilidad).....  | 49 |
| Tabla 9: Requisito no funcional de Portabilidad (instalabilidad)..... | 50 |
| Tabla 10: Descripción de autores. ....                                | 51 |
| Tabla 11: CU1: Autenticar Usuario. ....                               | 52 |
| Tabla 12: CU2: Mostrar contenido. ....                                | 53 |
| Tabla 13: CU3: Notificar el nivel.....                                | 53 |
| Tabla 14: CU4: Gestionar Usuario.....                                 | 53 |
| Tabla 15: CU5: Buscar Usuario. ....                                   | 54 |
| Tabla 16: CU6: Gestionar materiales.....                              | 54 |
| Tabla 17: CU7: Gestionar cuestionario.....                            | 54 |
| Tabla 18: CU8: Consultar evaluación.....                              | 54 |
| Tabla 19:CU9: Gestionar rutas.....                                    | 55 |

## Índice de figuras

|  |    |
|--|----|
| Figura 1: Interacción de los Módulos de un Sistema Tutor inteligente (Cataldi, 2007).....          | 9  |
| Figura 2: Esquema de un STI con sus módulos principales (cataldi, 2007). ....                      | 13 |
| Figura 3: Proceso del funcionamiento de una minería de datos (Davis, 2001).....                    | 16 |
| Figura 4: propuesta de solución del Sistema. Fuente:( Elaboración propia).....                     | 35 |
| Figura 5: Arquitectura de un sistema basado en reglas .....  | 36 |
| Figura 6: Modelo Conceptual utilizando el razonamiento basado en reglas (elaboración propia). .... | 39 |
| Figura 7: Casos de uso del sistema. ....   | 52 |
| Figura 8: Diagrama de clases Registrar Usuario .....   | 56 |
| Figura 9: Diagrama de clase de diseño Gestiona cuestionario (elaboración propia).....              | 57 |
| Figura 10: Modelo de datos (elaboración propia).....   | 58 |
| Figura 11: Diagrama de despliegue (elaboración propia). ....                                       | 59 |
| Figura 12: Esquema simplificado de la arquitectura MVP de Django (Condori, 2013). ....             | 60 |
| Figura 13: Diagrama de componentes (elaboración propia).....                                       | 63 |

## Introducción

En los últimos años las nuevas tecnologías de información y comunicación (TIC) han evolucionado rápidamente, logrando un salto vertiginoso en el desarrollo científico técnico a escala mundial. Y la inteligencia artificial (IA), surge como una de las ramas de estudios más recientes y promisorias en el campo de las ciencias de la computación, ella es una disciplina que estudia la creación y diseño de entidades capaces de razonar por sí mismas (Salgueiro, 2005).

Los intentos de utilizar estas tecnologías para favorecer el aprendizaje, comenzaron a finales de la década del 60. A partir de ese momento, la presencia de ordenadores en los hogares y en las escuelas ha tenido un crecimiento exponencial. Con el tiempo, la evaluación de su utilización daba una garantía de mejores resultados en el proceso de enseñanza-aprendizaje, dejando a su paso, una visión más prudente y exigente de la forma apropiada en la que se debe utilizar. Las innovaciones tecnológicas y la creciente popularidad y disponibilidad de la Internet, fueron razones principales para el desarrollo de numerosas aplicaciones y proyectos de investigación del uso de los medios informáticos en el campo de la tecnología educativa (Rodríguez, 2012).

Con la fuerte incursión de las tecnologías en la educación se han llevado a cabo múltiples investigaciones acerca de cómo poder establecer un ambiente de aprendizaje más interactivo que se acople a las necesidades cognitivas de los estudiantes. Se han desarrollado los sistemas tutores inteligentes (STI), que son sistemas de cómputo que se realizan con la finalidad de coadyuvar en el que hacer de las personas que fungen la tarea o funciones de tutor y así facilitar el aprendizaje de los estudiantes, que utiliza técnicas de inteligencia artificial para representar el conocimiento e interactúa con los estudiantes para enseñárselo (Hernández, 2015).

Los STI son programas que portan conocimientos sobre cierta materia y cuyo propósito es transmitir este conocimiento a los alumnos mediante un proceso interactivo individualizado, intentando simular la forma en que un tutor o profesor guiaría al alumno en el proceso de enseñanza-aprendizaje (Cataldi, 2010).

Los STI suministran aprendizaje en forma individualizada (Ambientes de Aprendizaje Individualizado), lo cual permite que el proceso de enseñanza aprendizaje sea más adaptable a las necesidades específicas o

nivel de aprendizaje del alumno. Para lograrlo, genera un modelo de aprendizaje basado en las Unidades Básicas de Aprendizaje y en los objetivos instruccionales, evalúa, detecta errores, brinda sugerencias, ejemplos, simulaciones y recomendaciones, y re planifica constantemente el modelo de aprendizaje; de la misma forma como lo hace un maestro humano (Builes, 2012).

Según Cataldi (2010): Los sistemas tutores inteligentes aplican algunas técnicas de la inteligencia artificial, esto con el objetivo de dotar el sistema de una habilidad que solo los seres humanos poseen, "inteligencia", contribuyendo así a que estos sistemas identifiquen las falencias en el aprendiz y puedan reforzar el conocimiento en el transcurso del aprendizaje de cierta área de conocimiento.

En las universidades y centros de enseñanza es palpable el incremento de la utilización de las TICs en el proceso de enseñanza-aprendizaje. En este contexto, las más demandadas son las tecnologías basadas en inteligencia artificial como los tutores inteligentes, los sistemas de gestión del aprendizaje y otros (Ferrer, 2017).

La Universidad de las Ciencias Informáticas (UCI) posee un alto grado de tecnología destinada a la labor docente-educativa, que sin dudas sirve de apoyo para el desarrollo de herramientas educativas que apoyen y mejoren el proceso de enseñanza-aprendizaje. Las herramientas educativas son programas diseñados con el propósito de apoyar la labor de los profesores en el proceso de enseñanza docente - educativo. Estas diferentes alternativas educativas adentran al estudiantado en el mundo de las tecnologías educativas.

El modelo de integración formación-producción-investigación implementado en la UCI permite la incorporación progresiva a la enseñanza semi presencial, con el uso del Sistema de Gestión de Cursos Moodle, el cual se utiliza a través del Centro Nacional de Educación a Distancia (CENED) y del Entorno Virtual de Aprendizaje (EVA), una plataforma en que profesores y estudiantes interactúan, propiciando así un aprendizaje colaborativo. Además el EVA mantiene organizada toda la información referente a las diversas asignaturas que componen el plan de estudio de la Universidad, incluyendo las asignaturas Inteligencia Artificial I (IA I) e Inteligencia Artificial II (IA II), que se ofertan en el 4to y 5to año de la carrera respectivamente. Sin embargo, el EVA y el CENED no cuentan con un mecanismo, que logre la personalización del proceso de enseñanza-aprendizaje y que se adapte a las necesidades cognitivas individuales de los estudiantes.

A pesar de todas las posibilidades que brinda la plataforma EVA establecida en la Universidad, los estudiantes que cursan la asignatura inteligencia artificial 1 presentan una serie de dificultades a la hora de la realización del estudio independiente, desconocen según sus características individuales, cuál sería la bibliografía más acertada para la apropiación del conocimiento, presentan deficiencias para autoevaluarse en una temática y la dificultad para ejercitar el contenido de la asignatura, pues no tienen el tiempo suficiente para consolidar el conocimiento de la misma ya que el profesor no está siempre disponible las 24 horas del día para aclarar las dudas puntuales que puedan surgir, siendo esto un factor que puede afectar los resultados en la en la asignatura. Además, como parte del contenido que se recibe en la asignatura de inteligencia artificial 1 se encuentra el lenguaje Prolog, el cual está basado en la lógica de primer orden. Este es uno de los lenguajes de programación que más trabajo les resulta a los estudiantes asimilar, debido a que le cambia el paradigma de programación con el que están familiarizado, más tomando en cuenta que para practicarlo hay que utilizar una aplicación por consola, a los cual no están acostumbrados.

Por otro lado, las herramientas y las alternativas de enseñanza que se utilizan hoy no tienen componente de inteligencia artificial; en este caso los sistemas basados en inteligencia artificial o inteligentes son creados con el objetivo de facilitar y agilizar el desempeño educativo, permitir hacer una atención individualizada del estudiante y que estén acorde con los gustos o intereses de los mismos. El hecho de que las herramientas educativas existentes hoy en la Universidad no presenten un componente inteligente es uno de los factores que atentan contra la motivación y el propio aprendizaje de los estudiantes.

Por todo lo anteriormente planteado se determina el siguiente **problema de investigación**:

¿Cómo contribuir al proceso de enseñanza-aprendizaje de la Inteligencia Artificial I, en la de la universidad de las Ciencias Informáticas?

Siendo identificado como **objeto de estudio**: Sistemas Tutoriales Inteligentes como apoyo al proceso de enseñanza-aprendizaje de la inteligencia artificial.

Para dar solución al problema antes descrito se tiene como **objetivo general**: Desarrollar los módulos tutor y dominio del Sistema Tutorial Inteligente para la enseñanza – aprendizaje de la Inteligencia Artificial I en la Universidad de las Ciencias Informáticas.

Se plantea como **campo de acción**: que la investigación está enmarcada en el modelado del tutor y dominio de un Sistema Tutorial Inteligente para la enseñanza de la Inteligencia Artificial I en la UCI.

Para darle cumplimiento al objetivo planteado se responderán las siguientes **preguntas científicas**:

- ¿Cuáles son los referentes teóricos que sustentan el desarrollo de los sistemas tutoriales inteligentes?
- ¿Qué aspectos deben tenerse en cuenta para realizar el análisis y diseño de los módulos tutor y dominio para el desarrollo de un STI como apoyo al proceso de enseñanza-aprendizaje de la Inteligencia Artificial I en la Universidad de las Ciencias Informáticas?
- ¿Cómo validar el adecuado análisis y diseño de los módulos tutor y dominio después de identificados todos los requisitos del sistema?

Acorde a las preguntas científicas ya expuestas, se trazan las siguientes **tareas de investigación**:

- Elaborar el marco teórico de la investigación relativa a los Sistemas Tutoriales Inteligentes para conocer su estructura y sus principales funcionalidades.
- Analizar de los principales referentes teóricos que sustentan el desarrollo y la utilización de los STI, haciendo especial énfasis en los módulos tutor y dominio.
- Describir las tecnologías, metodologías, lenguajes y herramientas utilizadas en el desarrollo de Sistemas Tutoriales Inteligentes.
- Identificar los componentes necesarios para el análisis y diseño de los módulos tutor y dominio de un STI en el proceso de enseñanza aprendizaje de la Inteligencia Artificial I.
- Validar el estudio del análisis y diseño de los módulos tutor y dominio de un STI

Con el cumplimiento de las tareas de investigación se espera como **posible resultado**: obtener el estudio del análisis y diseño para la posterior implementación de un sistema web, donde se apliquen técnicas de inteligencia artificial, el cual favorecerá al proceso de enseñanza - aprendizaje de la Inteligencia Artificial 1 en la UCI, ya que los estudiantes podrán aprender de una forma más dinámica y brindará una atención individualizada del estudiante más afín con sus características. Y que además permitirá guardar información

sobre la satisfacción de los estudiantes respecto a su trabajo en el sistema, y ayudará al estudio sistemático sin la total dependencia del profesor.

Para el desarrollo de la investigación se utilizaron los siguientes métodos científicos:

### **Métodos Teóricos:**

- **Analítico-Sintético:** Este método fue utilizado en la búsqueda de la bibliografía correspondiente al contenido sobre los módulos tutor y dominio de un Sistema Tutorial Inteligente (STI). Además, en el análisis de las diferentes técnicas de inteligencia artificial que se utilizan en el desarrollo de los STI, el estudio y descripción de las herramientas y tecnologías utilizadas para la posterior implementación del sistema y estudio de sistemas similares al marco de la investigación, permitiendo de esta manera la identificación de los elementos necesarios para dar solución al problema.
- **Análisis histórico-lógico:** Este método fue utilizado para el estudio crítico de las investigaciones anteriores, de su evolución y en el uso de estos como puntos de referencia y comparación de los resultados alcanzados. Este método permite entender la evolución y el surgimiento de los STI, así como las temáticas relacionadas con su implementación.
- **Modelación:** fue utilizado para la modelación de las funcionalidades de los módulos tutor y dominio, la representación explícita de la solución propuesta a través del análisis y diseño de los módulos, así como de los referentes teóricos extraídos de las fuentes bibliográficas consultadas, además para elaborar los diferentes modelos definidos en la metodología escogida.

### **Métodos Empíricos:**

- **Observación:** Este método fue utilizado para constatar las deficiencias existentes en los métodos pedagógicos utilizados en la UCI, así como investigar sobre el funcionamiento, ventajas y desventajas de otras aplicaciones educativas y las herramientas y metodologías más utilizadas que sirven para el análisis y diseño del tutor y dominio.
- **Entrevista:** Se realizan entrevistas con al menos 2 profesores de inteligencia artificial en la facultad 1 con el objetivo de conocer acerca de lo necesario para el desarrollo de los módulos tutor y dominio en el proceso de enseñanza-aprendizaje. Dicha entrevista se encuentra en los anexos 1.

**El presente documento está estructurado en dos capítulos:**

**Capítulo 1: Fundamentación teórica del análisis y diseño de los módulos tutor y dominio de un Sistema Tutorial Inteligente en el proceso de enseñanza - aprendizaje de la Inteligencia Artificial I:**

Este capítulo contiene la base teórica para entender el problema planteado, se realiza una valoración de la importancia de los módulos tutor y dominio del STI para la enseñanza y en la formación de futuros profesionales. Se describen la arquitectura y el funcionamiento del STI, así como los principales artefactos para el análisis y diseño de un STI, se argumenta el uso de las TIC en la educación caracterizando los Objetos de Aprendizaje como recursos de apoyo al Proceso de Enseñanza y Aprendizaje (PEA) de la inteligencia artificial. Finalmente se caracterizan las herramientas, lenguajes y metodologías estudiadas para la implementación.

**Capítulo 2: Diseño y análisis de la propuesta de solución para el desarrollo de los módulos tutor y dominio de un sistema tutorial inteligente para la enseñanza de la inteligencia artificial I:**

En este capítulo se muestra la propuesta de solución para la futura implementación de los módulos tutor y dominio de un STI definidas por la metodología Proceso Unificado Ágil (AUP, por sus siglas en inglés) variación UCI escenario 2. Se planifica el proceso de desarrollo del software. Se definen, entre otros, los casos de usos del sistema, los requisitos funcionales y no funcionales, el patrón arquitectónico y los patrones de diseño, los diagramas de diseño, el diagrama de despliegue y el diagrama de componentes.

# **CAPÍTULO 1. Fundamentación teórica del análisis y diseño de los módulos tutor y dominio de un Sistema Tutorial Inteligente en el proceso de enseñanza - aprendizaje de la Inteligencia Artificial I**

## **1.1- Introducción.**

En este capítulo se establecen conceptos fundamentales para la comprensión de la investigación. Se realiza un análisis del estado del arte de los modelos educativos que presentan características similares a los módulos que serán desarrollados. Se realiza una valoración de la importancia de los módulos tutor y dominio del STI y de las técnicas de inteligencia artificial (IA) más utilizadas. Se argumenta sobre el uso de las tecnologías en el proceso de enseñanza-aprendizaje, así como una breve descripción de las características y funcionalidades de los módulos. Finalmente se caracterizan las herramientas, lenguajes y metodologías consideradas para dar solución al problema de investigación.

## **1.2- Sistemas Tutores Inteligentes.**

Los sistemas tutores inteligentes (STI) comenzaron a desarrollarse en los años ochenta con la idea de poder impartir el conocimiento usando alguna forma de inteligencia para poder asistir y guiar al estudiante en su proceso de aprendizaje. Se buscó emular el comportamiento de un tutor humano, es decir a través de un sistema que pudiera adaptarse al comportamiento del estudiante, identificando la forma en que el mismo resuelve un problema a fin de poder brindarle ayudas cognitivas cuando lo requiera. (Cataldi, 2009)

A continuación, algunas definiciones de tutor inteligente que revelan sus orígenes y su esencia:

Según (VANLEHN, 1998) , define tutor inteligente como *un sistema de software que utiliza técnicas de inteligencia artificial (IA) para representar el conocimiento e interactúa con los estudiantes para enseñárselo.*

(Wolf, 1984), define los STI como: *“sistemas que modelan la enseñanza, el aprendizaje, la comunicación y el dominio del conocimiento del especialista y el entendimiento del estudiante sobre ese dominio.*

(Giraffa, 1997), los define como: *“un sistema que incorpora técnicas de IA (Inteligencia Artificial) a fin de crear un ambiente que considere los diversos estilos cognitivos de los alumnos que utilizan el programa”.*

Según (Wenger, 1997), lo define como *“Un STI que utiliza técnicas de inteligencia artificial principalmente para representar el conocimiento, y dirigir una estrategia de enseñanza, y que es capaz de comportarse como un experto tanto en el dominio del conocimiento que enseña (mostrando al alumno cómo aplicar dicho conocimiento), como en el dominio pedagógico donde es capaz de diagnosticar la situación en la que se encuentra el estudiante , y de acuerdo a ello , ofrecer una acción o solución que le permita progresar en el aprendizaje ”*.

Según (Cataldi, 2009), los STI permiten la emulación de un tutor humano para determinar qué enseñar, cómo enseñar y a quién enseñar a través de los siguientes módulos:

- **Módulo del Dominio:** define el dominio del conocimiento.
- **Módulo del Estudiante:** es capaz de definir el conocimiento del estudiante en cada punto durante la sesión de trabajo.
- **Módulo del Tutor:** genera las interacciones de aprendizaje basadas en las discrepancias entre el especialista y el estudiante.
- **Módulo Interfaz de Usuario:** permite la interacción del estudiante con un STI de una manera eficiente.

Las definiciones de los diversos autores anteriormente citados son de gran aporte a lo referente a la construcción de un STI.

Debido a la problemática planteada, para la investigación se basará en la definición del autor (Giraffa, 1997) ya que se desea hacer el análisis y diseño de un sistema educativo que incorpore técnicas de IA a fin de crear un ambiente que considere los diversos estilos cognitivos de los alumnos que utilizan el programa en este caso el STI. Se piensa en sentar las bases para un sistema de aprendizaje donde el estudiante le encuentre significado a sus acciones, que supere sus dificultades, incorporando conocimientos nuevos de un modo significativo y permanente.

A través de la interacción entre los módulos básicos, los STI son capaces de determinar lo que sabe el estudiante y cómo va en su progreso, por lo que la enseñanza se puede ajustar según las necesidades del estudiante, sin la presencia de un tutor humano.

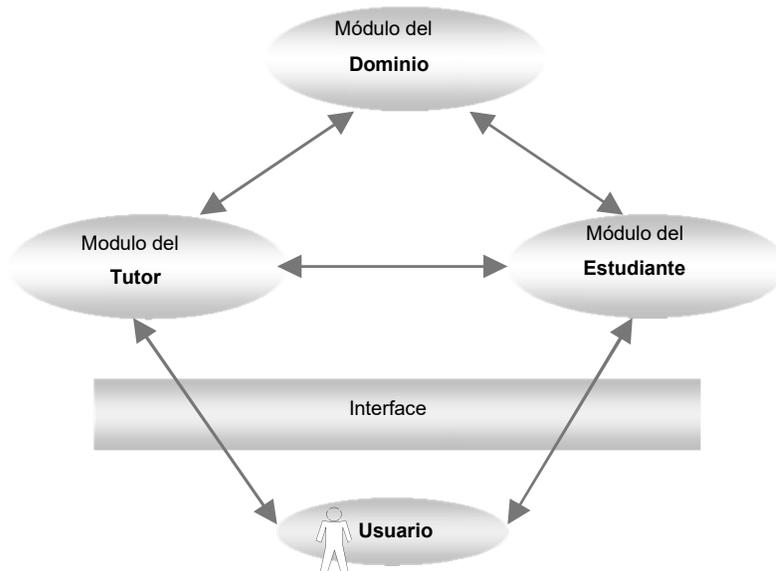


Figura 1: Interacción de los Módulos de un Sistema Tutor inteligente (Cataldi, 2007).

Siguiendo la idea de (Zulma Cataldi, 2009) los STI, responden a una arquitectura trimodular sin solapamiento de funcionalidades y con interfaces bien definidas a fin de obtener módulos independientes del dominio e intercambiables (Ver Figura 1).

**El módulo Estudiante:** tiene por objetivo realizar el diagnóstico cognitivo del alumno, y el modelado del mismo para una adecuada retroalimentación del sistema. Para el módulo estudiante se han planteado los siguientes sub-módulos.

1. **Estilos de aprendizaje:** Está compuesto por una base de datos con los estilos de aprendizajes disponibles en el sistema, los métodos de selección de estilos y las características de cada uno de ellos. Un estilo de aprendizaje es la forma de clasificar el comportamiento de un estudiante de acuerdo a la manera en que toma la información, forma las estrategias para aprender, cómo entiende y cómo le gusta analizar la información que está utilizando para acceder a un conocimiento determinado. En otras palabras, es una forma de agrupar o clasificar un estudiante de acuerdo a un perfil en relación con la información, ya que este estilo evoluciona y cambia de acuerdo a las variables de entorno y ambientales que afectan al estudiante.

- 2. Estado de conocimientos:** Contiene el mapa de conocimientos obtenido inicialmente a partir del módulo del dominio y que el actualizador de conocimientos irá modificando progresivamente a través de los resultados obtenidos en las evaluaciones efectuadas por el módulo del tutor, que será el encargado de enviar dichos resultados procesados.
- 3. Perfil psico-sociológico del estudiante:** Para determinar el perfil psico-sociológico se usa la Teoría de las Inteligencias Múltiples de Gardner (Gardner, 2001), donde se afirma que no existe una inteligencia única en el ser humano, sino una diversidad de inteligencias que evidencian las potencialidades y aspectos más significativos de cada individuo, en función de sus fortalezas y debilidades para la expansión de la inteligencia. Señala que las inteligencias trabajan juntas para: resolver problemas cotidianos, crear productos o para ofrecer servicios dentro del propio ámbito cultural.

**El módulo Dominio:** tiene el objetivo global de almacenar todos los conocimientos del campo de aplicación del STI. El dominio proporciona los conocimientos presentados en forma adecuada para que el alumno pueda adquirir las habilidades y conceptos requeridos, es decir, la capacidad de generar preguntas, explicaciones, respuestas y tareas, y además debe ser capaz de dar respuesta a los problemas y corregir las soluciones presentadas, analizando las diferentes aproximaciones válidas a la solución a través de la intervención del tutor. Se debe considerar qué tipo de conocimiento se está modelando según sea: declarativo, de procedimientos y cualitativo (Cataldi, 2009). Entre sus sub-módulos están los siguientes:

- 1. Parámetros básicos del sistema:** los cuales se almacenan en una base de datos.
- 2. Conocimientos:** son los contenidos que deben cargarse en el sistema, a través de los conceptos, las preguntas, los ejercicios, los problemas y las relaciones.
- 3. Elementos didácticos:** Son las imágenes, videos, sonidos, es decir material multimedia que se requiere para facilitarle al alumno apropiarse de conocimiento en la sesión pedagógica.

Generalmente el módulo dominio de un Sistema Tutorial Inteligente está compuesto por los tres submódulos citados anteriormente, siendo necesario que todos estén en el desarrollo del mismo para su perfecto funcionamiento. Por lo que estos formarán parte del módulo que se pretende desarrollar como propuesta de solución.

**El módulo Tutor:** es el encargado de definir y de aplicar la estrategia pedagógica de enseñanza (socrática, orientadora, dirigida etc.), de contener los objetivos a ser alcanzados y los planes utilizados para alcanzarlos. Es el responsable de seleccionar los problemas y el material de aprendizaje, de monitorear, y proveer asistencia al estudiante. También de integrar el conocimiento acerca del método de enseñanza, las técnicas didácticas y del dominio a ser enseñado (con integración de planificación y currículo). Es decir, un sistema de este tipo debe tratar además, los aspectos esenciales del currículo y de la planificación, ya que los aspectos de currículo involucran la representación, la selección y la secuenciación del material a ser utilizado y la planificación se refiere a cómo ese material va a ser presentado (Lage, 2009) .

Se piensa que un sistema que pueda emular al tutor humano y además que provea al estudiante de cierta flexibilidad para la selección del tipo de tutelado más adecuado, a sus preferencias, podría ser una solución factible para el problema planteado.

El mismo consta de los siguientes sub-módulos:

- 1. Protocolos pedagógicos:** estos son almacenados en una Base de Datos, con un gestor para la misma.
- 2. Planificador de lección:** se encarga de organizar los contenidos de la misma.
- 3. Analizador de perfil:** analiza las características del alumno, seleccionando la estrategia pedagógica más conveniente.

Generalmente el módulo tutor de un sistema tutorial inteligente está compuesto por tres submódulos citados anteriormente siendo necesario que todos estén en el desarrollo del mismo para su perfecto funcionamiento por lo que estos harán parte del módulo que se pretende desarrollar como propuesta de solución.

**Según (Cataldi, 2007) el objetivo del módulo tutor es:**

Mantener una jerarquía de metas que debe cumplir el STI mientras traspasa los conocimientos al alumno, quien producirá un resultado que el tutor no puede predecir de antemano. Para ello debe realizar las siguientes tareas:

- Identificar los objetivos para los alumnos, establece cuál es el objetivo de la sesión pedagógica que se está llevando a cabo y qué es lo que se pretende obtener como resultado para los estudiantes una vez finalizada dicha sesión.

- Ser capaz de detectar las necesidades del alumno, y, en base a estos requerimientos, modificar la estrategia de enseñanza.
- Supervisar y establecer el avance hacia los objetivos.
- Seleccionar los protocolos pedagógicos más eficientes para cada una de las sesiones pedagógicas, a fin de impartirla de tal manera que se logren los mejores resultados con los elementos disponibles. Estos elementos se obtienen a través del módulo del estudiante y del módulo del dominio.

Por otra parte, puntualmente en cada sesión debe:

- Presentar la explicación de un concepto (si es necesario de diferentes maneras, para variar el modo de presentar un tema si el alumno no alcanza a comprender el concepto nuevo).
- Exponer numerosos ejemplos sobre los conceptos analizados utilizando al agente docente y al generador de lenguaje natural.
- Recibir las respuestas del agente alumno a las evaluaciones y las eventuales consultas del alumno.
- Ordenar y actualizar el registro de los nuevos conceptos adquiridos por los alumnos.
- Supervisar las tareas que el alumno realice en colaboración con otros alumnos, como parte de su aprendizaje colaborativo.
- Comunicarse con los otros agentes Docentes, por ejemplo, para acordar tareas a realizar por los alumnos.

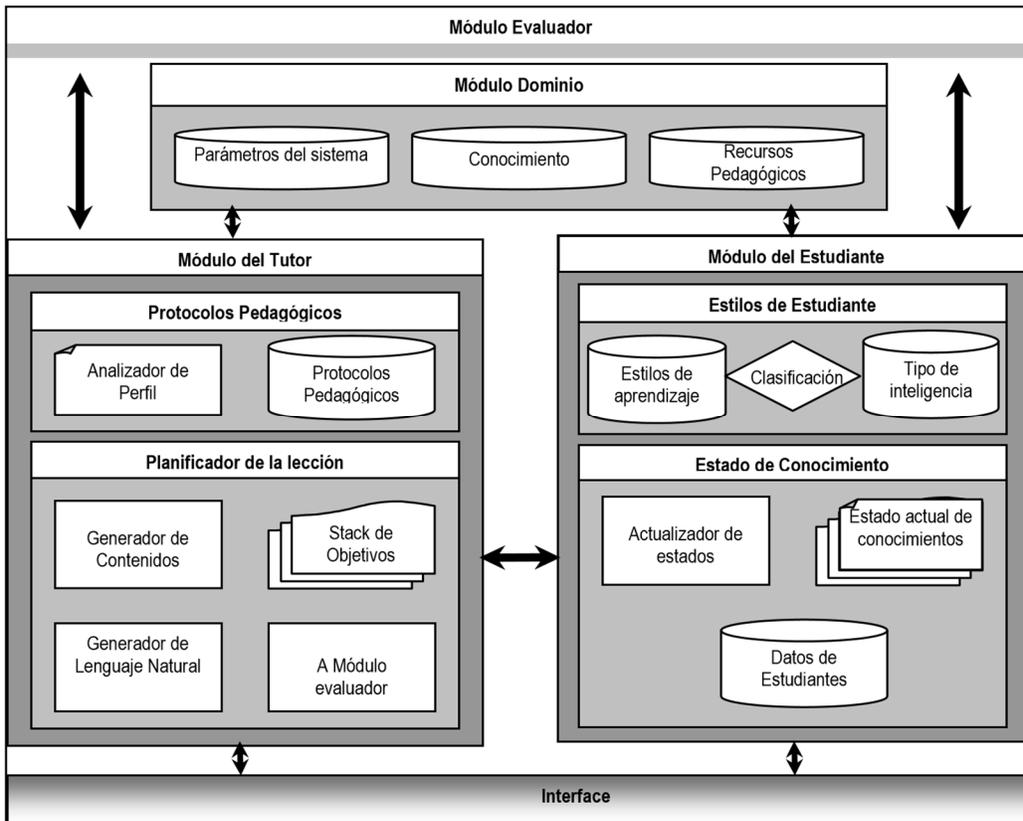


Figura 2: Esquema de un STI con sus módulos principales (cataldi, 2007).

### 1.3- Técnicas de Inteligencia Artificial usadas para el desarrollo de los módulos tutor y dominio de un STI.

A diferencia de la filosofía y la psicología que tratan de entender cómo funciona la inteligencia en abstracto, la inteligencia artificial, es un intento por descubrir y aplicar los aspectos de la inteligencia humana que pueden ser simulados mediante construcciones artificiales. Se observa que hasta las etapas tempranas de su desarrollo la inteligencia artificial ha presentado productos sorprendentes en sus aplicaciones (Salgueiro, 2011).

Hoy en día el campo de la inteligencia artificial enmarca varias sub-áreas tales como los sistemas expertos, la demostración automática de teoremas , el juego automático, el reconocimiento de voz y de patrones , el

procesamiento del lenguaje natural, la visión artificial, la robótica, las redes neuronales, entre otros. (Castillo, 1998).

En el contexto de los sistemas inteligentes se encuentra la utilización de técnicas de IA para determinar futuros rendimientos de los estudiantes, así como obtener el nivel de conocimiento asociado al estudiante, y para determinar patrones de aprendizaje de los mismos así como en la clasificación de los estudiantes, y en la asignación de bibliografías. Entre las técnicas de inteligencia artificial más utilizadas en los Sistemas Tutores Inteligentes encontramos las redes neuronales, las redes bayesianas y la minería de datos, entre otras (Zulma Cataldi, 2009). A continuación se describen algunas de estas técnicas.

### ➤ **Redes Neuronales**

En el contexto de los sistemas inteligentes se encuentran las redes neuronales, que son interconexiones masivas en paralelo de elementos simples y que responden a una cierta jerarquía intentando interactuar con los objetos reales tal como lo haría un sistema neuronal psicológico (Kohonen, 1988, 1998, 2001). Las redes neuronales poseen la característica de asimilar conocimiento en base a las experiencias mediante la generalización de casos, que las convierte en una herramienta interesante en el desarrollo de los modelados de la presente investigación.

Son modelos que intentan reproducir el comportamiento del cerebro. Del mismo modo que aquel, realiza una simplificación, averiguando cuáles son los elementos relevantes del sistema. Una elección adecuada de sus características, más una estructura conveniente, es el procedimiento convencional utilizado para construir redes capaces de realizar una determinada tarea. Este modelo posee dispositivos elementales de proceso, las neuronas, al igual que en el modelo biológico. El conjunto de estas pueden generar representaciones específicas, por ejemplo un número, una letra o cualquier otro objeto. (Ochoa, 2004).

Según el autor Ruiz, 2012, por ejemplo, para efectuar la predicción del rendimiento académico, se puede usar una red neuronal de tipo backpropagation tomando como datos de entrada los resultados de las evaluaciones parciales desagregados en dos formas.

- Tomando el caso de resolución por ejercicios.
- Tomando ejercicios en función de los logros cognitivos.

Usando los datos provenientes de las evaluaciones parciales de los estudiantes se pueden predecir futuros rendimientos.

### ➤ **Redes Bayesianas**

Las redes bayesianas son herramientas estadísticas orientadas a la inferencia probabilística y en el ámbito de la tutorización electrónica se pueden utilizar para modelar la incertidumbre asociada al estudiante y su nivel de conocimientos. Los algoritmos genéticos, se fundamentan en el concepto biológico de la evolución natural y son utilizados en procesos de optimización. Se fundamentan en los mecanismos de la selección natural, por los que sólo sobreviven los individuos más aptos, luego de la interacción entre los mismos, pertenecientes a una población de posibles soluciones (Davis, 1999).

### ➤ **Minería de Datos**

Nos encontramos inmersos en una época donde el volumen de datos generados está creciendo en volumen y a un ritmo relevante, de ahí la importancia que cobra la minería de datos. Y es que, la evolución de la tecnología y la capacidad de computación nos permiten adquirir conocimiento y ventajas competitivas mediante su explotación a través de la minería.

La **minería de datos** es la ciencia o metodología que permite explotar los datos con el objetivo de generar modelos que posibiliten describir, encontrar patrones, establecer agrupaciones, clasificar, segmentar o asociar productos, clientes o cualquier otra entidad objeto de obtener conocimiento y ser aplicados a otros nuevos obteniendo respuestas de acción inmediata (Davis Falkenauer, 2001).

Busca generar información similar a la que podría generar un experto humano, que además satisfaga el principio de comprensibilidad. El objetivo de éste es descubrir conocimientos interesantes; como patrones, asociaciones, cambios, anomalías y estructuras significativas a partir de grandes cantidades de datos almacenados en bases de datos, data warehouse, o cualquier otro medio de almacenamiento de información.

La **Minería de Datos** es un caso especial del **Aprendizaje Automático**, utiliza sus métodos para encontrar patrones, con la diferencia que el escenario observado es una base de datos. En un esquema de Aprendizaje Automático, el mundo real es el entorno sobre el cual se realiza el aprendizaje, estos se

traducen en un conjunto finito de observaciones u objetos que son codificados en algún formato legible. El conjunto de ejemplos constituye la información necesaria para el entrenamiento del sistema.

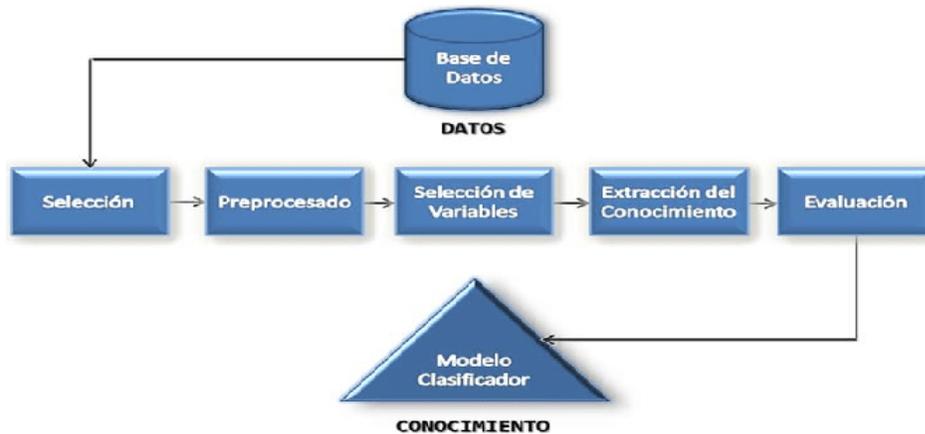


Figura 3: Proceso del funcionamiento de una minería de datos (Davis, 2001).

### ➤ **Sistemas basados en el conocimiento (SBC)**

Los sistemas basados en el conocimiento (SBC) constituyen técnicas de IA válidas para enfrentar la construcción de un STI, dada por sus aspectos afines. Pero no todos los paradigmas para crear SBC, facilitan la concepción de un STI, donde lo fundamental para su desarrollo es determinar cómo representar el conocimiento requerido para sus módulos y a partir de dicho conocimiento realizar un diagnóstico del estudiante para que el sistema se adapte a sus características. Sin embargo, similitudes de los STI y los SBC son factores a estudiar para concebir todos los módulos de los STI y un diagnóstico adecuado del qué y cómo enseñar dependiendo del estudiante (Natalia Martínez Sánchez, 2011).

### ➤ **Sistemas Basados en Reglas (SBR)**

Un sistema basado en reglas (SBR) es un sistema basado en el conocimiento en el que se hace una representación mediante reglas de producción o reglas condicionales. Las reglas representan el conocimiento utilizando el formato SI-ENTONCES (IF-THEN).

SI (IF), es el antecedente, premisa, condición o situación.

ENTONCES (THEN), es el consecuente, conclusión, acción o respuesta.

En este tipo de sistema, la base de conocimiento (BC) contiene las variables y el conjunto de reglas que definen el problema, y la máquina de inferencia (MI) obtiene las conclusiones aplicando la lógica a estas reglas. La BC contiene la representación del conocimiento sobre el dominio de aplicación del sistema y se divide en la base de hechos y en la base de reglas. La base de hechos representa el conocimiento en las variables de entrada y salida del sistema y en la base de reglas se combinan variables de la base de hechos con el IF y el THEN junto con los operadores lógicos AND y OR. La parte de una regla que se encuentra entre IF y THEN es el antecedente y el resto el consecuente, en el antecedente es donde están las premisas que se deben cumplir para que la regla sea aplicable y en el consecuente el conjunto de acciones derivadas de la aplicación de la regla. La MI utiliza los datos y el conocimiento para obtener conclusiones o hechos. Por ejemplo, si la premisa de una regla es cierta, entonces la conclusión de la regla también debe ser cierta (José Manuel Gutiérrez, 2016).

Para obtener estas conclusiones se utilizan diferentes tipos de reglas y estrategias de inferencia y control. Entre las reglas de inferencia se encuentran Modus Ponens y Modus Tollens y en cuanto a las estrategias de inferencia, el encadenamiento de reglas y el encadenamiento de reglas orientado a un objetivo, cuyas reglas son utilizadas por la MI para obtener conclusiones simples, que son las que resultan de una regla simple y las conclusiones compuestas, que son las que resultan de más de una regla.

Los SBR emplean el Encadenamiento hacia adelante, en el que los nuevos hechos son inferidos de hechos existentes y el Encadenamiento hacia atrás, por ejemplo, si hay una regla IF A THEN B, aplicar encadenamiento hacia atrás significa usar el conjunto de reglas existentes para determinar que debe ser cierto para que B también lo sea. Esta técnica es empleada para probar una hipótesis.

### ➤ **Sistemas Basados en Casos (SBC)**

Los sistemas basados en casos son los que más se asemejan al modo de pensar de los seres humanos. Diariamente, al enfrentarse a nuevas situaciones, lo primero que se hace es buscar en la memoria experiencias anteriores similares y a partir de allí, se establecen semejanzas y diferencias que se ajustan a las soluciones dadas anteriormente para obtener una nueva solución.

El razonamiento basado en casos (RBC) es una técnica de IA que intenta llegar a la solución de nuevos problemas de forma similar a como lo hacen los seres humanos, utilizando la experiencia acumulada hasta el momento en acontecimientos similares. Entre los componentes fundamentales que contiene este tipo de sistema, se encuentran la BC, el módulo de recuperación de casos y el módulo de adaptación de las soluciones. Los “casos” son problemas resueltos y almacenados en la base de conocimiento BC. Cuando hay un nuevo problema que resolver, éste es descrito para el módulo de recuperación, el cual realiza una búsqueda en la BC y encuentra problemas o casos similares. Estos problemas o casos similares resueltos son recuperados (caso recuperado) y enviados al módulo de adaptación, donde son analizados para construir una solución para el nuevo problema y una vez hallada la solución se almacena junto con la descripción del problema en la BC, ya que constituye un nuevo caso (Natalia Martínez Sánchez, 2011).

### **1.3.1- Fundamentos de la selección**

Aunque el sistema basado en casos es la estrategia de razonamiento que más se asemeja al modo de pensar de los seres humanos, y son los esquemas más comúnmente utilizados para la representación del conocimiento, no es conveniente aplicar este tipo de razonamiento, pues la forma que se utiliza de descomponer el conocimiento del dominio y generalizarlo en casos, resultaría un proceso complejo que no se ajusta al modelo que se quiere desarrollar. Además, se haría un análisis detallado y un poco engorroso de los rasgos y de las semejanzas de cada estudiante para establecer la estrategia pedagógica a seguir. Las Redes Neuronales tampoco se utilizan pues no es objetivo desarrollar modelos no lineales inspirado en el funcionamiento del cerebro, las redes bayesianas tampoco se utilizan porque son herramientas estadísticas orientadas a la inferencia probabilística y trabajan con la incertidumbre, se debe destacar que en ambas técnicas es necesario tener conocimiento almacenado de situaciones similares a la problemática que se desea resolver, cuestión con que no se cuenta para el desarrollo de la investigación. Por otro lado, la minería de datos es utilizada en proyectos donde se requiere el almacenamiento de una gran cantidad de datos, cuestión que en estos momentos no es necesaria para la solución de la problemática planteada.

Sin embargo el sistema basado en reglas cuenta con la experiencia de los expertos, que permiten modelar su conocimiento a partir de reglas de producción sencillas, para seleccionar los protocolos pedagógicos a utilizar en la caracterización de los estudiantes por eso la selección de esta técnica de IA.

## **1.4- Análisis de sistemas similares**

Algunas tecnologías como los agentes inteligentes, las redes neuronales, los sistemas expertos, los sistemas tutores inteligentes, que hace unos años solo existían en forma teórica o en los ámbitos universitarios, se utilizan a diario en aplicaciones de uso relativamente sencillo que se encuentran en la etapa de producción y no son solo casos de estudio en los laboratorios universitarios. Además estas tecnologías están disponibles para las grandes empresas y los centros de alta tecnologías, así como también para aplicaciones pequeñas y medianas que pueden utilizarlas a diario (Salgueiro, 2010).

A finales del siglo XIX, se realizaron estudios sistemáticos, basados en distintas ciencias, como la psicología, la educación, la sociología, la medicina, etc. Para explicar el proceso de aprendizaje y el funcionamiento de la mente humana. Se crearon muchas teorías a partir de estas investigaciones, desde las fisiológicas que explican el funcionamiento del cerebro humano en función de intercambio de neuroreceptores y diferencias de potencial hasta las filosóficas que intentan explicar el funcionamiento del cerebro humano a partir de los estímulos externos (Salgueiro, 2010).

### **1.4.1- Análisis a nivel Internacional**

Los primeros STI siguieron las ideas de (Carbonell, 1970) y los paradigmas de programación convencional. De entre ellos se pueden destacar: Scholar, Why, Sophie, Guidon, West, Buggy, Debuggy, Steamer, Meno, Proust, Sierra (Cataldi, 2010).

Entre los STI más recientes se destacan:

#### **SircSim**

Fue desarrollado en conjunto por el departamento de ciencias de la computación del instituto de Illinois Instituto de Tecnología y el departamento de Fisiología de Rush Colegio de Medicina. Este tutor es el más avanzado actualmente en su tipo y se utiliza para complementar las clases teóricas sobre problemas cardiovasculares está constituido por los cuatro módulos de un STI, los módulos dominio, dominio, estudiante y la interfaz, utiliza redes neuronales como técnicas de inteligencia artificial (Cataldi, 2010).

#### **AGT (Advancced Geometry Tutor)**

Es un STI para el uso en clases de geometría avanzada, desarrollado en la Universidad de Pittsburgh a través de la National Science Foundation through y el Centro de Interdisciplinary Research on Constructive Learning Environments de la Universidad de Pittsburgh y Carnegie Mellon University. Se basa en comparar dos estrategias de resolución de problemas, encadenamiento hacia adelante y encadenamiento hacia atrás para determinar el aprendizaje de los estudiantes sobre teorema de la geometría, con el fin de determinar qué estrategia acelera más el aprendizaje, está compuesto por los cuatro módulos de un STI como técnica de inteligencia artificial utiliza redes neuronales (Cataldi, 2010).

### **Sistema tutorial inteligente para el tratamiento de los errores gramaticales para el español como Lengua Extranjera con fines académicos**

Este sistema tiene como objetivo principal el diseño y la implementación de un analizador automático de errores gramaticales para el español en el contexto de un tutorial inteligente para la enseñanza del español como Lengua Extranjera (ELE). Con el objetivo de lograr un sistema que incorpore técnicas de PLN (procesamiento de lenguaje natural), que sea capaz de reconocer de manera efectiva los errores gramaticales que efectúen los alumnos y dar una respuesta útil y específica, es necesario construir un analizador sintáctico (*parser*) para el tratamiento de dichos errores que opere en un entorno virtual y que dé respuestas automáticas. Otra característica importante de este *parser* es que dará respuestas diferentes según el nivel de aprendizaje en que se encuentre el alumno. Está constituido por los cuatro módulos de un STI, los módulos dominio, dominio, estudiante y la interfaz (Anita Ferreira Cabrera, 2007).

### **Tutor Inteligente con reconocimiento y manejo de emociones para Matemáticas**

Es un Sistema Tutor Inteligente para matemáticas de tercer grado de primaria que identifica el estado emocional del estudiante y produce retroalimentación afectiva para el mismo durante un curso, el cual se encuentra instalado en una red social. La red social y el Sistema Tutor Inteligente con manejo afectivo han sido probados en escuelas públicas y privadas de la localidad, con resultados muy favorables. El reconocimiento de emociones se lleva a cabo a través de expresiones faciales, lo cual se realiza por medio de una red neuronal artificial como técnica de inteligencia artificial. El sistema tutor inteligente y afectivo para la red social adopta el modelo tradicional conocido como arquitectura de cuatro-módulos, donde el primer módulo (la interfaz de la red social) tiene acceso a otros tres módulos principales llamados: dominio, estudiante y tutor (Lourdes Ferrer, 2017).

## **STI para el apoyo de la Enseñanza de la Lectura Inicial**

Se diseñó y desarrolló un software educativo para apoyar el proceso de aprendizaje de la lectura inicial, basado en los conocimientos actuales sobre enseñanza de la lectoescritura y en el modelo de Sistemas Instruccionales Inteligentes. El Software desarrollado se evaluó en una muestra de 8 niños de una escuela municipal de Santiago de Chile, identificados por la profesora como presentando algún grado de retraso lector. Se evaluaron los siguientes aspectos: claridad de las instrucciones, utilidad de los refuerzos y ayudas, sensibilidad al nivel de desempeño del niño y potencial como herramienta instruccional. Está compuesto por el Modelo del dominio, Modelo del Alumno y Modelo Pedagógico. Se utilizó como técnica de Inteligencia Artificial un Sistemas Basados en Reglas para regular el flujo de actividades asociadas a cada niño (Luis Lauzardo, 2017).

### **1.4.2- Análisis a nivel nacional**

En el mundo está muy difundido el uso de los STI. En Cuba existen muy poco de ellos demostrándose así la poca difusión y uso de este tipo de medios educativos en el país. Entre los más destacados están los siguientes:

#### **El Sistema Tutorial Inteligente para Diagnóstico y Tratamiento de Infecciones de Transmisión Sexual (STIITS):**

Fue desarrollado en la universidad de Cienfuegos en conjunto con especialistas de segundo grado del hospital Dr. Gustavo Aldereguia Lima. Que hicieron labor de expertos en la validación de la base de conocimientos para el posterior diagnóstico de las enfermedades, así como aportar todos los datos para la realización del tutorial. STIITS responde a una necesidad social de la provincia de Cienfuegos, y también nacional por sus posibilidades de empleo en la docencia. No existe algún otro software en el país con características similares para el mismo fin, está constituido por los cuatro módulos de un STI, los módulos dominio, dominio, estudiante y la interfaz y utiliza razonamiento basado en casos como técnica de inteligencia artificial (Maribel García, 2012).

## **APA-Prolog**

Es un ambiente de enseñanza-aprendizaje para la programación lógica que se adapta a los conocimientos previos del estudiante, elaborado por el grupo de desarrollo de Software para la Educación de la Universidad de Sancti Spíritus “José Martí Pérez”. Su interfaz la constituye un conjunto de mapas conceptuales relacionados entre sí y utiliza Razonamiento Basado en Casos (RBC) como técnica de Inteligencia Artificial.

### 1.4.3- A nivel UCI

#### **VIRTEVALL**

Fue desarrollado en 2009 en la facultad 3 de la universidad de las ciencias informáticas, este STI prevé crear un ambiente alternativo de evaluación para el aprendizaje autónomo de idiomas extranjeros, que favorezca el trabajo de los estudiantes de manera independiente en sus debilidades, y profundizar en temas que sean de su interés en el aprendizaje del idioma inglés y utiliza el razonamiento basado en casos como técnica de inteligencia artificial.

#### **STI-SO**

Desarrollado en la Universidad de las Ciencias Informáticas, es una herramienta de apoyo al proceso de enseñanza-aprendizaje de la asignatura Sistema Operativo, la cual ayuda a consolidar los contenidos del plan de estudio definido para la asignatura. El STI-SO fue estructurado en cuatro módulos: Módulo del Estudiante, Módulo del Dominio, Módulo Pedagógico y la Interfaz. Este STI utiliza el RBC como técnica de IA para asignarle al estudiante el material de estudio ( Luisa Idorka Morales , 2012).

*Tabla 1: Estudio de los sistemas similares.*

| <b>STI</b> | <b>Características/Funcionalidades</b> | <b>Técnicas de inteligencia artificial(IA)</b> | <b>Utilidad</b> | <b>Alcance</b> | <b>Multiplataforma</b> |
|------------|--|--|-----------------|----------------|------------------------|
|------------|--|--|-----------------|----------------|------------------------|

|   |  |                                   |   |   |      |
|---|--|-----------------------------------|---|---|------|
| SircSim   | se utiliza para complementar las clases teóricas sobre problemas cardiovasculares  | Redes Neuronales                  | X | X | X    |
| AGT   | un STI para el uso en clases de geometría avanzada, desarrollado en la Universidad de Pittsburgh   | Redes neuronales                  | X | X | ---- |
| STI para el tratamiento de los errores gramaticales para el español | como objetivo principal el diseño y la implementación de un analizador automático de errores gramaticales para el español en el contexto de un tutorial inteligente para la enseñanza del Español como Lengua Extranjera | procesamiento de lenguaje natural | X | X | ---- |
| STI- de reconocimiento y manejo de emociones para Matemáticas       | Sistema para matemáticas de tercer grado de primaria que identifica el estado emocional del estudiante y produce retroalimentación afectiva para el mismo durante un curso   | Redes Neuronales                  | X | X | ---- |
| STIITS  | validación de la base de conocimientos para el diagnóstico de las enfermedades, de transmisión sexual  | Razonamiento basado en casos      | X | X | X    |

|  |   |                               |   |   |      |
|--|---|-------------------------------|---|---|------|
| STI - SO   | Es una herramienta de apoyo al proceso de enseñanza-aprendizaje de la asignatura Sistema Operativo, la cual ayuda a consolidar los contenidos del plan de estudio definido para la asignatura.                    | Razonamiento basado en casos  | X | X | ---- |
| Virtevall  | prevé crear un ambiente alternativo de evaluación para el aprendizaje autónomo de idiomas extranjeros,  | Razonamiento basado en casos  | X |   | ---- |
| APA - Prolog   | Es un Ambiente de enseñanza-aprendizaje para la programación lógica que se adapta a los conocimientos previos del estudiante,   | Razonamiento Basado en casos  | X | X | X    |
| STI- para el Apoyo de la Enseñanza de la Lectura Inicial | software educativo para apoyar el proceso de aprendizaje de la lectura inicial, basado en los conocimientos actuales sobre enseñanza de la lectoescritura y en el modelo de Sistemas Instruccionales Inteligentes | Razonamiento basado en reglas | X | X | ---- |

#### **1.4.4- Valoración de los sistemas similares**

Con la investigación realizada a cada uno de los sistemas antes expuestos se pudo constatar que todos ayudan en diferentes áreas de la educación y la gran utilidad de estos sistemas. De forma general se puede afirmar que casi todos ellos en su mayoría están compuestos por los cuatro componentes principales de un STI que son el módulo dominio, el módulo tutor el módulo del estudiante y la interfaz, y aunque no se puede usar ninguno de ellos ya que no se relacionan exactamente con el problema propuesto y no cumplen con las necesidades específicas del trabajo ya que no son sistemas destinados al proceso de enseñanza de la inteligencia artificial , su análisis ayudo en la búsqueda e identificación de las características y elementos necesarios a tener en cuenta para dar solución la propuesta de solución. Las técnicas de inteligencia artificial utilizadas por los sistemas analizados son: Los Sistemas Basados en Casos, los Basados en Reglas, las Redes Neuronales, y el Procesamiento de Lenguaje Natural. Siendo la más utilizada el razonamiento basados en casos. Para la presente investigación se propone utilizar los SBR pues cuenta con la experiencia de los expertos, que permiten modelar su conocimiento a partir de reglas de producción sencillas, para seleccionar los protocolos pedagógicos a utilizar en la caracterización de los estudiantes por eso la selección de esta técnica de IA.

#### **1.5- Metodologías de desarrollo de software y tecnologías a utilizar**

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y un soporte documental para el desarrollo de productos de software. Indicando paso a paso las actividades a realizar para lograr el producto informático deseado, indicando qué personas deben participar en el desarrollo de las actividades y qué papel deben jugar. Además detallan la información necesaria para realizar una actividad y qué se debe producir como resultado de su ejecución. (Pressman, 2005).

Para el desarrollo de la solución propuesta se seleccionó Proceso Unificado Ágil (AUP, por sus siglas en inglés), versión UCI escenario 2, debido a que logra estandarizar el proceso de desarrollo de software, dando cumplimiento a las buenas prácticas que define CMMI-DEV3 v1.3. Se logra hablar un lenguaje común en cuanto a fases, disciplinas, roles y productos de trabajos.

### 1.5.1- Proceso Unificado Ágil (AUP)

El Proceso Unificado Ágil de Scott Ambler o Agile Unified Process (AUP) en inglés es una versión simplificada del Proceso Unificado de Rational (RUP). Este describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP (Tamara Rodríguez Sánchez, 2015).

### 1.5.2- Fases de la metodología

De las 4 fases que propone AUP (Inicio, Elaboración, Construcción, Transición) se decide para el ciclo de vida de los proyectos de la UCI mantener la fase de Inicio, pero modificando el objetivo de la misma, se unifican las restantes 3 fases de AUP en una sola, a la que llamaremos Ejecución y se agrega la fase de Cierre. Para una mayor comprensión se describen cada una de ellas (Tamara Rodríguez Sánchez, 2015).

**Inicio:** Durante el inicio del proyecto se llevan a cabo las actividades relacionadas con la planeación del proyecto. En esta fase se realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcance del proyecto, realizar estimaciones de tiempo, esfuerzo y costo y decidir si se ejecuta o no el proyecto.

**Ejecución:** En esta fase se ejecutan las actividades requeridas para desarrollar el software, incluyendo el ajuste de los planes del proyecto considerando los requisitos y la arquitectura. Durante el desarrollo se modela el negocio, obtienen los requisitos, se elaboran la arquitectura y el diseño, se implementa y se libera el producto.

**Cierre:** En esta fase se analizan tanto los resultados del proyecto como su ejecución y se realizan las actividades formales de cierre del proyecto.

### 1.5.3- Características por escenarios

**Escenario No 1:** Proyectos que modelen el negocio solo pueden modelar el Sistema con Caso de Uso de Sistema (CUS).

**Escenario No 2:** Proyectos que modelen el negocio con Modelo Conceptual (MC), solo pueden modelar el sistema con Casos de Uso del Sistema (CUS).

**Escenario No 3:** Proyectos que modelen el negocio con Descripción de Proceso de Negocio (DPN), solo pueden modelar el sistema mediante la Descripción de Requisitos por Proceso (DRP).

**Escenario No 4:** Proyectos que no modelen el negocio, solo pueden modelar el sistema con Historias de Usuario (HU).

Para la presente investigación, se escoge el escenario 2, pues se aplica a los proyectos que hayan evaluado el negocio a informatizar y como resultado obtengan que no es necesario incluir las responsabilidades de las personas que ejecutan las actividades, de esta forma modelarían exclusivamente los conceptos fundamentales del negocio. Se recomienda este escenario para proyectos donde el objetivo primario es la gestión y presentación de información.

#### **1.5.4- Fundamentos de la selección**

Se escoge la metodología AUP UCI porque describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos. Es un marco de desarrollo de software iterativo e incremental. A menudo es considerado como un proceso altamente ceremonioso porque especifica muchas actividades y artefactos involucrados en el desarrollo de un proyecto de software.

#### **1.6- Lenguajes de Programación**

Un lenguaje de programación es un conjunto de símbolos y reglas sintácticas y semánticas que se utiliza para escribir los programas de computadoras. Puede clasificarse teniendo en cuenta distintos criterios: lenguajes interpretados, lenguajes compilados, lenguajes de script, entre otros. El programador debe ser capaz de seleccionar el lenguaje de programación apropiado para resolver un problema bajo ciertas circunstancias específicas, de manera óptima. Actualmente existen un conjunto de lenguajes de programación tanto para plataformas GNU/Linux, como para Windows, entre los que se destacan, python, Java, C++, entre otros (Language de Programación, 2019).

### 1.6.1- Python v2019

Python es un lenguaje interpretado de alto nivel y orientado a objetos. El enfoque de Python está en la legibilidad y la velocidad. Su sintaxis simple y fácil de usar enfatiza la legibilidad, su amplia biblioteca de módulos y paquetes conlleva la velocidad de desarrollo y la orientación a objetos fomenta la modularidad del programa y la reutilización del código. Al ser un lenguaje de código abierto, tiene una gran comunidad y es ampliamente adoptado. Gracias a esto se ha convertido en uno de los lenguajes de más rápido crecimiento. Python también admite visualizaciones de datos y trazado basados en datos reales (Sevilla, Fernández, y Díaz, 2017).

#### Características Generales del lenguaje.

- **Legible:** Python está diseñado para ser sencillo, elegante, consistente y similar a las matemáticas. Tiene una sintaxis consistente y se asemeja bastante al lenguaje humano y matemático hasta el punto de ser escrito y leído como si fuera pseudocódigo.
- **Multiplataforma:** lo cual es ventajoso para hacer ejecutable su código fuente entre varios sistemas operativos.
- **Fuertemente tipado:** significa que el tipo de valor no cambia repentinamente. Un *string* que contiene solo dígitos no se convierte mágicamente en un número. Cada cambio de tipo requiere una conversión explícita.

### 1.6.2- Fundamentos de la selección

Se escoge python como lenguaje de programación porque python es para todos, sus características funcionales trabajan como un puente entre los académicos y los ingenieros provienen de campos matemáticos y estadísticos no tienen gran experiencia en la creación de código pero las facultades de legibilidad de python posibilitan un rápido aprendizaje e implementación en los proyectos que desarrollan y actualmente es el lenguaje de programación que más se utiliza para el desarrollo de proyectos relacionados con la inteligencia artificial. Además es la base de desarrollo de muchas compañías y el número de productos y proyectos basados en python está creciendo a escala exponencial.

### 1.6.3. Lenguaje marcado de hipertexto (HTML 5)

Es un nuevo lenguaje de marcado de hipertexto para presentar y estructurar el contenido en internet. Es la quinta revisión y nueva versión del estándar HTML. HTML5 ofrece nuevas características que proporcionan no sólo un rico soporte multimedia (video y audio), sino que también mejoran el apoyo para la creación de aplicaciones web. Esta nueva tecnología puede proporcionar una solución rentable para implementar aplicaciones para apoyar diferentes dispositivos (por ejemplo, ordenador, tableta, teléfono inteligente personal) sin tener que construir aplicaciones diferentes para cada tipo de dispositivos (Gauchat, 2014).

### 1.7- Entorno de desarrollo Integrado (IDE)

Un Entorno de Desarrollo Integrado es un programa compuesto por una serie de herramientas que utilizan los programadores para desarrollar código. Esta herramienta puede estar pensada para su utilización con un único lenguaje de programación o bien puede dar cabida a varios de estos. Las herramientas que normalmente componen un entorno de desarrollo integrado son las siguientes: un editor de texto, un compilador, un intérprete, unas herramientas para la automatización, un depurador, un sistema de ayuda para la construcción de interfaces gráficas de usuario y opcionalmente, un sistema de control de versiones. (Rodríguez., 2010).

#### 1.7.1- PyCharm v9.2.1.

PyCharm en su versión 2019.2.1, es un entorno de desarrollo integrado multiplataforma utilizado para desarrollar en el lenguaje de programación *Python*. Provee funcionalidades que permiten una experiencia única y aumentan en gran medida la productividad ((JetBrains, 2016):

- Completamiento de código de manera inteligente y señalamiento de errores con reparación de los mismos de forma automatizada.
- Integración con marcos de trabajo de desarrollo web modernos como *Django*.
- Depurador integrado y herramientas de pruebas.
- Soporte para bases de datos e integración con sistemas de control de versiones.

## 1.8- Marco de Trabajo.

Según el autor Rodríguez .O ,2010: un marco de trabajo es una estructura de archivos y utilidades que aceleran la programación de una aplicación informática, proveyendo una metodología de trabajo que sistematiza y facilita la generación de formularios, funciones y módulos de uso común, permitiendo al desarrollador dedicar su atención hacia los aspectos específicos de cada aplicación .

En el desarrollo de software un entorno o marco de trabajo es una estructura conceptual y tecnológica de asistencia definida, normalmente, con artefactos o módulos concretos de software, que puede servir de base para la organización y desarrollo de software. Típicamente, puede incluir soporte de programas, bibliotecas, un lenguaje interpretado entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

### 1.8.1- Django v1.8

Se seleccionó *Django* como framework de desarrollo web del lado del servidor, pues, es construido sobre *Python*, fomenta el desarrollo ágil permitiendo la construcción de aplicaciones de alta calidad en un breve período de tiempo. Utiliza una modificación de la arquitectura Modelo – Vista – Controlador (MVC) llamada Modelo – Plantilla – Vista (*MTV*, por sus siglas en inglés). *Django* impulsa el desarrollo de código limpio y promueve la utilización de buenas prácticas de desarrollo web como el principio *DRY*. Entre algunos casos de éxito se encuentran: *Instagram*<sup>9</sup>, *Pinterest*<sup>10</sup> y *The New York Times* (Condori, 2013).

## 1.9- Lenguaje de Modelado

Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language) es una de las mejores herramientas para analizar y diseñar sistemas de software que ofrece un lenguaje común que todo desarrollador debe conocer. Es un conjunto de herramientas que permite modelar (analizar y diseñar) sistemas orientados a objetos (UML, 2018).

### 1.9.1. Herramienta para el modelado

Para modelado del tutor se hará uso de una herramienta de Ingeniería de Software Asistida por Computadoras (CASE) por sus siglas en inglés. Las herramientas CASE son aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas ayudan en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el diseño de proyectos, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras (Somerville, 2010).

### 1.9.2- Visual Paradigm v8.0

Se selecciona como herramienta CASE de modelado profesional, que utiliza UML para la completa representación de las etapas por las que transita un producto de software. Este permite la realización de una amplia gama de diagramas como: casos de uso, de actividades, de despliegue, entre otros, así como la generación de código fuente desde los mismos y la documentación asociada al proceso que esté siendo modelado (Visual Paradigm, 2013).

### 1.10- Sistema Gestor de Base de Datos

Se define un Sistema Gestor de Bases de Datos o SGBD, también llamado DBMS (Data Base Management System) como una colección de datos relacionas entre sí, estructurados y organizados, y un conjunto de programas que acceden y gestionan esos datos.

Se trata de un conjunto de programas no visibles al usuario final que se encargan de la privacidad, la integridad, la seguridad de los datos y la interacción con el sistema operativo. Proporciona una interfaz entre los datos, los programas que los manejan y los usuarios finales. Cualquier operación que el usuario hace contra la Base de Datos está controlada por el gestor.

Como objetivos principales de los SGBD constan los siguientes:

- **Independencia de datos:** Los programas de aplicación deben verse afectados lo menos posible por cambios efectuados en datos que no usan.

- **Integridad de los datos:** La información almacenada en la BD debe cumplir ciertos requisitos de calidad, para ello hace falta, en el momento de introducirse los valores de los datos, que éstos se almacenen debidamente, y que posteriormente no se deterioren.
- **Seguridad de los datos:** A la información almacenada en la BD sólo pueden acceder las personas autorizadas y de la forma autorizada.

Se determinó que el Gestor de Base de Datos (SGBD) para almacenar toda la información generada por el STI será MySQL en su versión 5.7.

### 1.10.1- MySQL v5.7

Es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation y está considerada como la base datos open source más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web (MySQL, 2018).

A continuación algunas características esenciales de MySQL:

- Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Posibilidad de selección de mecanismos de almacenamiento que ofrecen diferentes velocidades de operación, soporte físico, capacidad, distribución geográfica y transacciones.
- Transacciones y claves foráneas.
- Conectividad segura
- Replicación.
- Soporta gran cantidad de tipos de datos para las columnas.
- Flexible sistema de contraseñas (passwords) y gestión de usuarios, con un muy buen nivel de seguridad en los datos.

### **1.12- PhpMyAdmin v4.8.3**

Es una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web, utilizando un navegador web. Actualmente puede crear y eliminar Bases de Datos , crear, eliminar y alterar tablas , borrar, editar y añadir campos , ejecutar cualquier sentencia SQL, administrar claves en campos, administrar privilegios, exportar datos en varios formatos y está disponible en 72 idiomas. Se encuentra disponible bajo la licencia GPL Versión 2 (García, 2008).

### **1.13- Conclusiones del capítulo**

En este capítulo se describieron los principales resultados obtenidos de la búsqueda de los principales referentes teóricos que sustentan la investigación, a partir de lo cual el autor concluye que:

A partir de la descripción del análisis y diseño de los módulos tutor y dominio de un STI se obtuvo una visión general de cómo funcionan actualmente estos sistemas y de qué forma contribuyen al proceso de enseñanza aprendizaje.

El estudio de sistemas similares realizado demostró que existen varias tipos de STI destinados al proceso de enseñanza y varias herramientas usadas para su desarrollo , pero que no son factibles a la problemática planteada ya que no cumple con las características específicas de lo que se requiere. No obstante, este estudio posibilitó identificar características útiles a tener en cuenta en la propuesta de solución.

La selección de la metodología seleccionada es adecuada para solución propuesta haciendo posible la recopilación de los principales artefactos y requisitos que se necesitan para la implementación futura de la solución propuesta.

## **Capítulo 2: Diseño y análisis de la propuesta de solución para el desarrollo de los módulos tutor y dominio de un sistema tutorial inteligente para la enseñanza de la inteligencia artificial I**

### **2.1- Introducción**

En este capítulo se presentan las fases de ejecución y cierre definidas por la metodología *AUP* variación UCI escenario 2. Se realiza la propuesta de solución y se planifica el proceso de desarrollo de software, identificando los requisitos funcionales y no funcionales de la propuesta de solución, patrones de diseño y estilo arquitectónico. Se realiza una breve descripción lógica de los módulos Tutor y Dominio del STI así como de la técnica de IA seleccionada.

### **2.2- Propuesta de Solución**

Con el objetivo de influir positivamente en el aprendizaje de la inteligencia artificial 1 en los estudiantes, quedó definido en el capítulo anterior el conjunto de herramientas a utilizar para el desarrollo del STI. Se pretende desarrollar un sistema que garantice un eficiente desarrollo del conocimiento de los estudiantes con respecto al estudio de la inteligencia artificial.

Para cumplimentar el objetivo propuesto al inicio de esta investigación, los estudiantes se identificarán en el sistema y una vez validado que sus datos sean correctos podrán realizar un cuestionario para determinar su estilo de aprendizaje, siendo este el paso principal para su inserción en la aplicación. El sistema tiene como objetivo que el estudiante domine las habilidades orientadas en los estudios independientes o tareas de la asignatura inteligencia artificial 1, las cuales serán definidas mediante un tema en específico y orientados según su característica o forma de percibir los conocimientos.

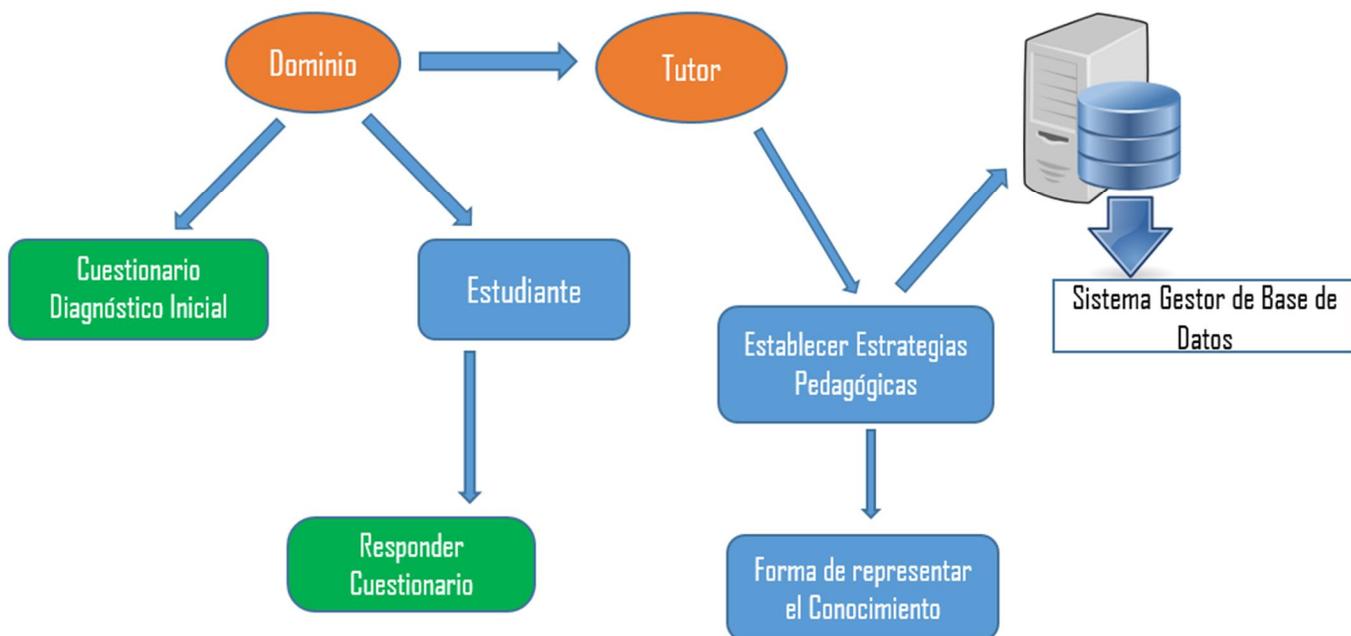


Figura 4: propuesta de solución del Sistema. Fuente:( Elaboración propia).

### 2.2.1- Cuestionario Diagnóstico

Una vez que el estudiante entra al sistema debe responder un cuestionario diagnóstico a través del cual se sabrá el nivel que posee el mismo en cuanto a su conocimiento en IA 1. Debe existir una biblioteca de esta asignatura con artículos designados por los profesores. Una tarea asignada por el profesor a los estudiantes que dependa de uno de los artículos de la biblioteca.

### 2.2.2- Indicadores y Descriptores de la dimensión Acceso a la Información:

**Búsqueda de la información:** Usa fuentes confiables y pertinentes para encontrar la información de acuerdo con la necesidad determinada para resolver la tarea en cuestión.

**Selección de la información:** Elije los documentos digitales para resolver la necesidad de información.

**Filtrado:** (fecha de publicación, fuente reconocida sobre la temática y actualidad del contenido del artículo.

### 2.2.3- Medida de Valor para la calificación del diagnóstico:

Excelente

Bien

Regular

Después de haber diagnosticado al estudiante enviarle un mensaje al profesor.

El estudiante que obtuvo los niveles parcialmente bien o regular debe trabajar en un curso referente a los resultados alcanzados en el examen bajo el asesoramiento del sistema; el que le facilitará tarjetas ayuda, ejemplos y ejercicios que le permitirán alcanzar un nivel superior en caso de no superar estos niveles con la ayuda del sistema tendrán que trabajar bajo el asesoramiento del profesor de la asignatura.

Al comienzo del trabajo se definió el razonamiento basado en reglas como técnica de IA para dar solución a la problemática y al desarrollo de la propuesta de solución. A continuación se presenta cómo funciona esta técnica.

### 2.3. Sistema Basado en reglas

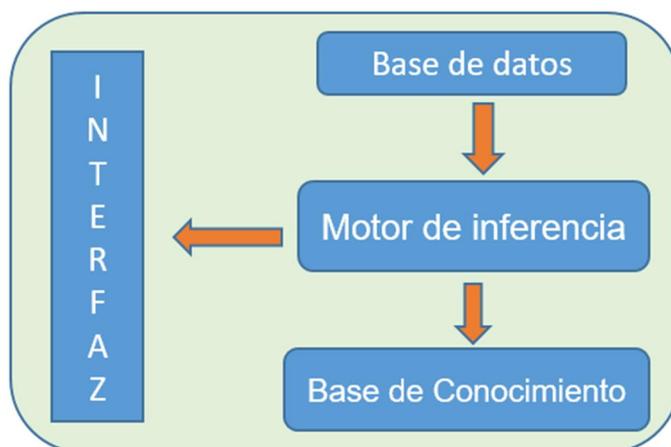


Figura 5: Arquitectura de un sistema basado en reglas

**Base de Datos:** Contiene los datos sobre un problema que se han descubierto durante una consulta. Durante una consulta con el sistema experto, el usuario introduce la información del problema actual en la base de datos. El sistema empareja esta información con el conocimiento disponible en la base de casos para deducir los nuevos datos. La base de datos pertenece al módulo estudiante, es el que almacena toda la información que se posee del estudiante.

**Base de Conocimiento:** Es la parte del sistema experto que contiene el conocimiento sobre el dominio. Hay que obtener el conocimiento del experto y codificarlo en la base de conocimiento. Una forma clásica de representar el conocimiento en un sistema experto son las reglas que en este caso serían las características de los estilos de aprendizaje del estudiante y como el mismo recibirá el conocimiento. La base de conocimiento pertenece al módulo dominio que es donde se encuentran todas las reglas y bibliografías del sistema.

**Interfaz:** La interacción entre un sistema experto y un usuario se realiza en lenguaje natural. Es la que se encarga de recibir la información necesaria del usuario para el sistema.

**Motor de Inferencia:** Trabaja con la información contenida en la base de conocimiento y la base de datos para deducir nuevos conocimientos. Contrasta los datos particulares de la base de datos con el conocimiento contenido en la base de conocimiento para obtener conclusiones acerca del problema. El motor de inferencia pertenece al módulo tutor, este módulo se encarga de asignarle el protocolo pedagógico al estudiante. A partir de los hechos, que son las características del estudiante almacenadas en el módulo estudiante y utiliza la base de conocimiento, que son las reglas las cuales se encuentran almacenadas en el módulo dominio.

## 2.4- Modelo Conceptual

Un modelo conceptual es una representación visual en forma de diagrama de las clases conceptuales u objetos del mundo real que son significativos en un dominio de interés; no se trata de un conjunto de diagramas que describen clases u objetos de software con responsabilidades (Sommerville, 2010).

Los modelos conceptuales pueden utilizarse para capturar y expresar el entendimiento ganado en un área bajo análisis como paso previo al diseño de un sistema, ya sea de software o de otro tipo. Similares a los mapas mentales utilizados en el aprendizaje, el modelo conceptual es utilizado por el analista como un medio para comprender el sector industrial o de negocios al cual el sistema va a servir (Sommerville, 2011).

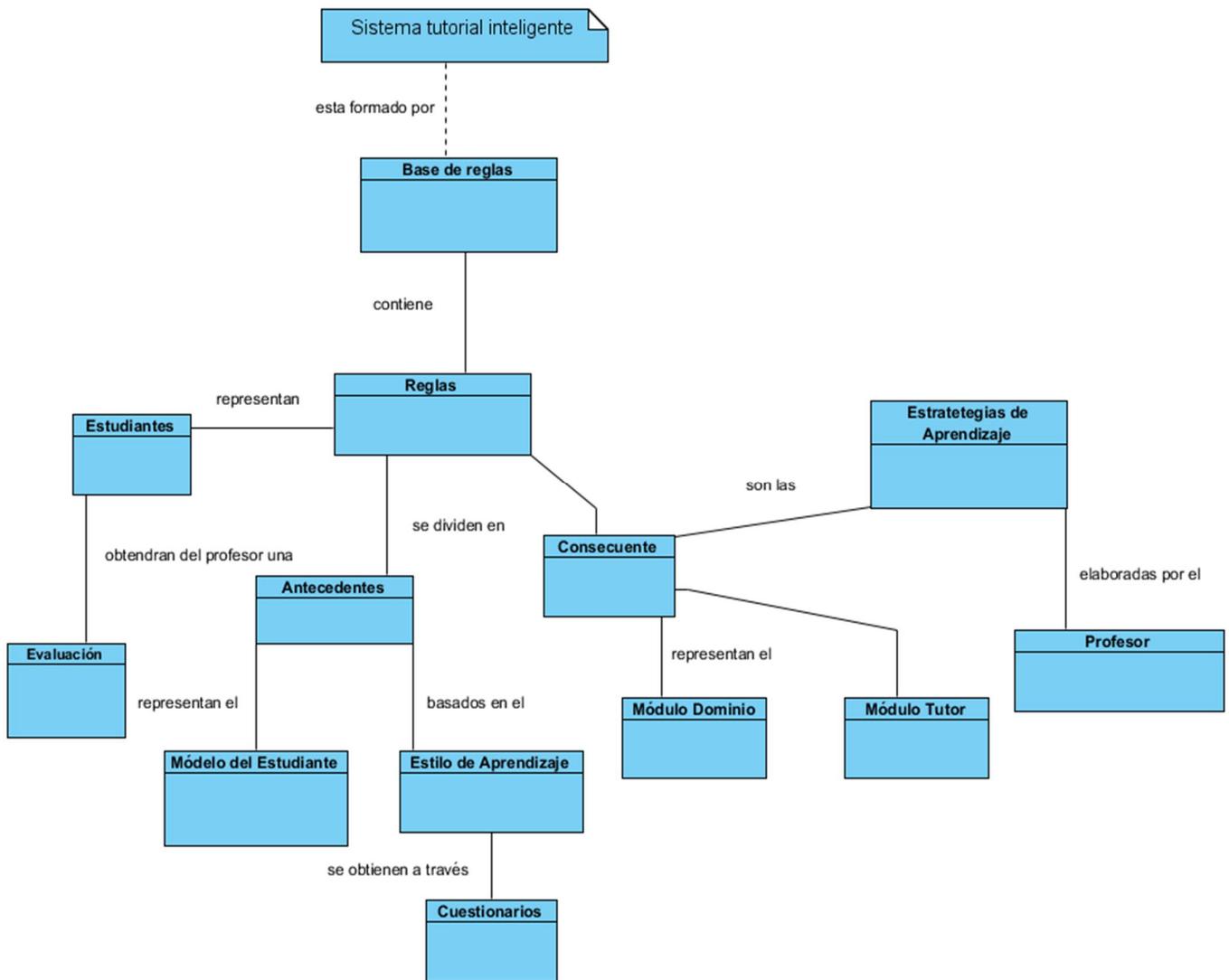


Figura 6: Modelo Conceptual utilizando el razonamiento basado en reglas (elaboración propia).

## 2.5- Personas relacionadas con el sistema

La siguiente tabla muestra la descripción de las personas relacionadas con el sistema que son todas aquellas que obtienen un resultado del mismo.

Tabla 2: Personas relacionadas con el sistema.

| Personas relacionadas con el sistema | Responsabilidades   |
|--------------------------------------|---|
| Usuario                              | Es cualquier persona que accede a la aplicación, se autentica y el sistema le da la posibilidad de realizar cuestionarios.  |
| Profesor                             | Es aquel usuario al cual el Administrador otorga este rol, teniendo privilegios de gestionar documentos, gestionar cuestionarios y consultar los resultados de los estudiantes. |
| Administrador                        | Es la persona encargada de gestionar el sistema administrando las cuentas de los usuarios registrados en la aplicación y gestionando el contenido publicado en el mismo.        |
| Estudiante                           | Es el usuario al que le asignan los protocolos pedagógicos  |

## 2.6- Lista de Reserva de Productos.

Según define la metodología AUP, la Lista de Reserva del producto (LRP) contiene los requisitos funcionales ordenados según la prioridad de implementación y los requisitos no funcionales del sistema.

Según Martínez e Imelda (2015): un requisito es simplemente una declaración abstracta de alto nivel de un servicio que debe proporcionar el sistema o una restricción de éste. En el otro extremo es una definición detallada y formal de una función del sistema.

### **2.6.1- Requisitos funcionales**

Un requisito funcional define una función del sistema de software o sus componentes. Una función es descrita como un conjunto de entradas, comportamientos y salidas. Los requisitos funcionales pueden ser cálculos, detalles técnicos, manipulación de datos y otras funcionalidades específicas que se supone que un sistema debe cumplir (Sommerville, 2010).

Seguidamente se especifican los requerimientos del sistema correspondientes.

**RF1.** Autenticar usuario

**RF2.** Buscar Usuario

**RF3.** Modificar Datos del usuario

**RF4.** Eliminar Usuario.

**RF5.** Crear Cuestionario Diagnóstico.

**RF6.** Actualizar Cuestionario Diagnóstico.

**RF7.** Revisar Cuestionario Diagnóstico

**RF8.** Mostrar Resultados Diagnóstico.

**RF9.** Mostrar Datos del Usuario.

**RF10.** Mostrar Contenido.

**RF11.** Notificar Nivel del Estudiante.

**RF12.** Mostrar Materiales de Aprendizaje.

**RF13.** Actualizar Materiales de Aprendizaje

**RF14.** Crear Rutas de Aprendizaje.

**RF15.** Aplicar Rutas de Aprendizaje.

**RF16.** Cambiar Rutas de Aprendizaje.

**RF17.** Consultar evaluación.

A continuación la descripción de cada uno de los requisitos funcionales

*Tabla 3: Descripción de los requisitos funcionales*

| Nombre                           | Descripción   | Complejidad | Prioridad |
|----------------------------------|---|-------------|-----------|
| RF1. Autenticar usuario          | El usuario debe autenticarse para ser identificado. Si es el Administrador, puede gestionar usuario, gestionar materiales, gestionar cuestionario y consultar evaluaciones de otros usuarios. Si está registrado como Profesor, tendrá los mismos privilegios que el Administrador excepto gestionar usuario, de lo contrario, solo podrá realizar cuestionarios. | Alta        | Alta      |
| RF2. . Buscar Usuario            | Buscar en el sistema la información de cada usuario.  | Baja        | Media     |
| RF3. Modificar Datos del usuario | El administrador tiene el poder de manipular y modificar la información del usuario.  | Media       | Alta      |
| RF4. Eliminar Usuario            | Permite al administrador y el profesor eliminar estudiante o un usuario.  | Baja        | Baja      |

|   |  |       |       |
|---|--|-------|-------|
| RF5. Crear Cuestionario Diagnóstico.      | El profesor creará el cuestionario para esto el alumno llenara los campos correspondientes a las preguntas que se realizarán.                                  | Media | Media |
| RF6. Actualizar Cuestionario Diagnóstico. | Permite al profesor modificar y actualizar el estado de los usuarios   | Alta  | Media |
| RF7. Revisar Cuestionario Diagnóstico     | Permite al profesor revisar los resultados del cuestionario Diagnóstico.   | Alta  | Alta  |
| RF8. Mostrar Resultados Diagnóstico.      | Permite al profesor visualizar el resultado del Cuestionario Diagnóstico.  | Media | Baja  |
| RF9. Mostrar Datos del Usuario.           | Permite al administrador Mostrar los datos y todas las acciones realizadas por el usuario después de autenticarse y acceder al sistema.                        | Media | Media |
| RF10. Mostrar Contenido.                  | El sistema deberá permitir mostrar todas las presentaciones de los contenidos impartidos mediante diapositivas, video clases, entre otros recursos educativos. | Baja  | Media |
| RF11. Notificar Nivel del Estudiante.     | El sistema deberá notificar el nivel del estado de aprendizaje del estudiante.   | Media | Media |
| RF12. Mostrar Materiales de Aprendizaje.  | El administrador tiene la tarea de visualizar en el sistema todos los materiales de aprendizaje que se utilizarán.   | Baja  | Media |

|  |  |       |       |
|--|--|-------|-------|
| RF13. Actualizar Materiales de Aprendizaje | Permite al administrador manipular toda la información de los documentos que tengan acceso a la aplicación, o sea, el administrador puede, eliminar y modificar documentos.        | Alta  | Alta  |
| RF14. Crear Rutas de Aprendizaje.          | Permite al profesor crear rutas de aprendizaje de acuerdo a las características de los estudiantes.  | Alta  | Alta  |
| RF15. Aplicar Rutas de Aprendizaje.        | Después de evaluadas todas las rutas se selecciona la más adecuada para cumplir con el proceso de aprendizaje.   | Media | Media |
| RF16.Cambiar Rutas de Aprendizaje.         | El profesor tiene el privilegio de cambiar o modificar la ruta de aprendizaje si la ruta seleccionada anteriormente no cumple con los requisitos o no satisface a los estudiantes. | Alta  | Media |
| RF17. Consultar evaluación.                | Después de aplicadas todas las rutas y todo el proceso de aprendizaje el profesor debe consultar la evaluación de los estudiantes y establecer la nota para cada uno.              | Media | Media |

## 2.6.2- Requisitos no funcionales

Los requisitos no funcionales o atributo de calidad, en la ingeniería de sistemas y la ingeniería de software, son aquellos que no se refieren directamente a las funcionalidades específicas que proporciona el sistema, sino a las propiedades de éste ( Sommerville, 2006).

### Software

- **RnF1:** Utilización del navegador web Mozilla Firefox como aplicación predeterminada y conexión de red.

### Hardware

- **RnF2:** Se definen las características que debe tener el entorno de trabajo que son la cantidad de memoria RAM (8GB) CPU (4 x 2.50 GHz (Intel (R) core (TM) i5-7th Gen, Windows 10 pro, capacidad de disco duro de 64 a utilizar en los clientes, el servidor, y como servidor de bases de datos MySQL.

### Usabilidad

- **RnF3:** Facilidad de uso: el sistema debe tener una interfaz amigable que permita el uso sencillo de la aplicación, además de acceder a la información de manera efectiva y rápida de forma que los usuarios con poca experiencia se adapten fácilmente.

### Seguridad

- **RnF4:** La seguridad está dada por la **Confidencialidad** de los datos al autenticarse el usuario, pues la contraseña es de conocimiento personal. Y la **Integridad** ya que el sistema debe garantizar que la información se mantenga inalterable durante todo el proceso de uso del mismo.

## Interfaz

- **RnF5:** Las ventanas del sistema: contienen claro y bien estructurados los datos, además de permitir la interpretación correcta de la información.
- **RnF6:** Interfaz: Se emplearán colores cómodos a la vista, sin acumulación de imágenes u objetos que distraigan al usuario de su objetivo. Permitirá a los usuarios interactuar con el sistema de forma fácil.

## Portabilidad

- **RnF7** El sistema debe ser adaptable a diferentes entornos especificados.
- **RnF8:** El sistema debe poderse instalar en un entorno especificado.

A continuación, se describirán los requisitos no funcionales que debe poseer el sistema:

*Tabla 4: Requisito no funcional de Software*

| Atributo de Calidad                                | Software   |
|--|--|
| Sub-atributos/Sub-características                  | Garantizar una buena conexión con navegador Mozilla Firefox como aplicación predeterminada |
| Objetivo   | Lograr una conexión sin fallos   |
| Origen   | Interno al sistema   |
| Artefacto  | Servicios del sistema (cliente y servidor) / Canales de comunicación                       |
| Estímulo   |  |
| 1.a Interrupción de comunicaciones de red en la PC | <b>Respuesta: Flujo de eventos (Escenarios)</b>  |

|  |   |
|--|---|
|  | Tratar de conectarse cada cierto tiempo para terminar el proceso de comunicación / Continuar funcionando en modo normal |
| <b>Medida de Respuesta</b>             |   |
| NA                                     |   |
| <b>2.a Inactividad del cliente</b>     |   |
|  | El servidor se encargará de notificar los clientes inactivos / Continuar funcionando en modo normal                     |
| <b>Medida de Respuesta</b>             |   |
| NA                                     |   |
| <b>3.a Ocurrencia de una excepción</b> |   |
|  | Se notifica al usuario / Continuar funcionando en modo normal   |
| <b>Medida de respuesta</b>             |   |
| NA                                     |   |

*Tabla 5: Requisito no funcional de Usabilidad (comprensión)*

|                            |            |
|----------------------------|------------|
| <b>Atributo de Calidad</b> | Usabilidad |
|----------------------------|------------|

|  |  |
|--|--|
| <b>Sub-atributos/Sub-características</b> | Comprensión  |
| <b>Objetivo</b>                          | Lograr que el sistema sea entendible fácilmente para los usuarios. |
| <b>Origen</b>                            | El usuario   |
| <b>artefacto</b>                         | El sistema   |
| <b>Entorno</b>                           | <b>El sistema funciona correctamente</b>                           |
| <b>Estímulo</b>                          | <b>Respuesta: Flujo de eventos (Escenarios)</b>                    |
| <b>NA</b>                                | NA   |
| <b>Medida de respuesta</b>               |  |
| <b>NA</b>                                |  |

Tabla 6: Requisito no funcional de Usabilidad (Operabilidad)

|  |   |
|--|---|
| <b>Atributo de Calidad</b>               | Usabilidad  |
| <b>Sub-atributos/Sub-características</b> | Operabilidad  |
| <b>Objetivo</b>                          | Lograr que el sistema sea capaz de realizar todas las operaciones solicitadas por el cliente. |
| <b>Origen</b>                            | El usuario  |
| <b>Artefacto</b>                         | El sistema  |
| <b>Entorno</b>                           | El sistema está funcionando correctamente.  |
| <b>Estímulo</b>                          | <b>Respuesta: Flujo de eventos (Escenarios)</b>   |
| NA                                       | NA  |
| <b>Medida de respuesta</b>               |   |

NA

Tabla 7: Requisito no funcional de Hardware

|  |  |
|--|--|
| <b>Atributo de Calidad</b>               | Hardware   |
| <b>Sub-atributos/Sub-características</b> | Utilización de recursos  |
| <b>Objetivo</b>                          | CPU (4 x 2.50 GHz (Intel (R) core(TM) i5 – 7thGen<br>RAM (8GB)<br>Windows 10 pro<br>SO – 64 bits, procesador x64 |
| <b>Origen</b>                            | Externo al sistema   |
| <b>Artefacto</b>                         | Servidor de almacenamiento de información  |
| <b>Entorno</b>                           | Operación normal   |
| <b>Estímulo</b>                          | Respuesta: Flujo de eventos (Escenarios)   |
| <b>NA</b>                                | NA   |

Tabla 8: Requisito no funcional de Portabilidad (adaptabilidad)

|  |               |
|--|---------------|
| <b>Atributo de Calidad</b>               | Portabilidad  |
| <b>Sub-atributos/Sub-características</b> | Adaptabilidad |

|  |  |
|--|--|
| <b>Objetivo</b>  | Lograr que el sistema sea adaptable a diferentes entornos especificados. |
| <b>Origen</b>  | El usuario   |
| <b>Artefacto</b>   | El sistema   |
| <b>Entorno</b>   | Funcionando correctamente  |
| <b>Estímulo</b>  | <b>Respuesta: Flujo de eventos (Escenarios)</b>                          |
| <b>1. a Se prueba el sistema en diferentes entornos.</b> | El sistema funciona correctamente  |
| <b>Medida de respuesta</b>                               |  |
| NA   |  |

Tabla 9: Requisito no funcional de Portabilidad (instalabilidad)

|  |   |
|--|---|
| <b>Atributo de Calidad</b>               | Portabilidad  |
| <b>Sub-atributos/Sub-características</b> | Instalabilidad  |
| <b>Objetivo</b>                          | Lograr que el sistema se pueda instalar en un entorno especificado. |
| <b>Origen</b>                            | El usuario  |
| <b>Artefacto</b>                         | El sistema  |
| <b>Entorno</b>                           | Funcionando correctamente   |

|  |   |
|--|---|
| <b>Estímulo</b>                                    | <b>Respuesta: Flujo de eventos (Escenarios)</b> |
| 1. a Se instala el sistema en diferentes entornos. |   |
|  | El sistema funciona correctamente.              |
| <b>Medida de respuesta</b>                         |   |
| NA   |   |

## 2.7- Casos de Uso del Sistema

El diagrama de casos de uso representa la forma en cómo un Cliente (Actor) opera con el sistema en desarrollo, además de la forma, tipo y orden en cómo los elementos interactúan (Pressman, 2005).

A continuación, se describen los actores y los casos de uso del sistema.

*Tabla 10: Descripción de autores.*

| <b>Actor</b>  | <b>Descripción</b>   |
|---------------|--|
| Administrador | Responsable por gestión de los usuarios del sistema, así como la gestión de los materiales de aprendizaje. |
| Profesor      | Gestiona el cuestionario, las tareas de aprendizaje y establece las rutas de aprendizaje.                  |
| Sistema       | Muestra el contenido disponible así como en nivel de acceso de los estudiantes.                            |

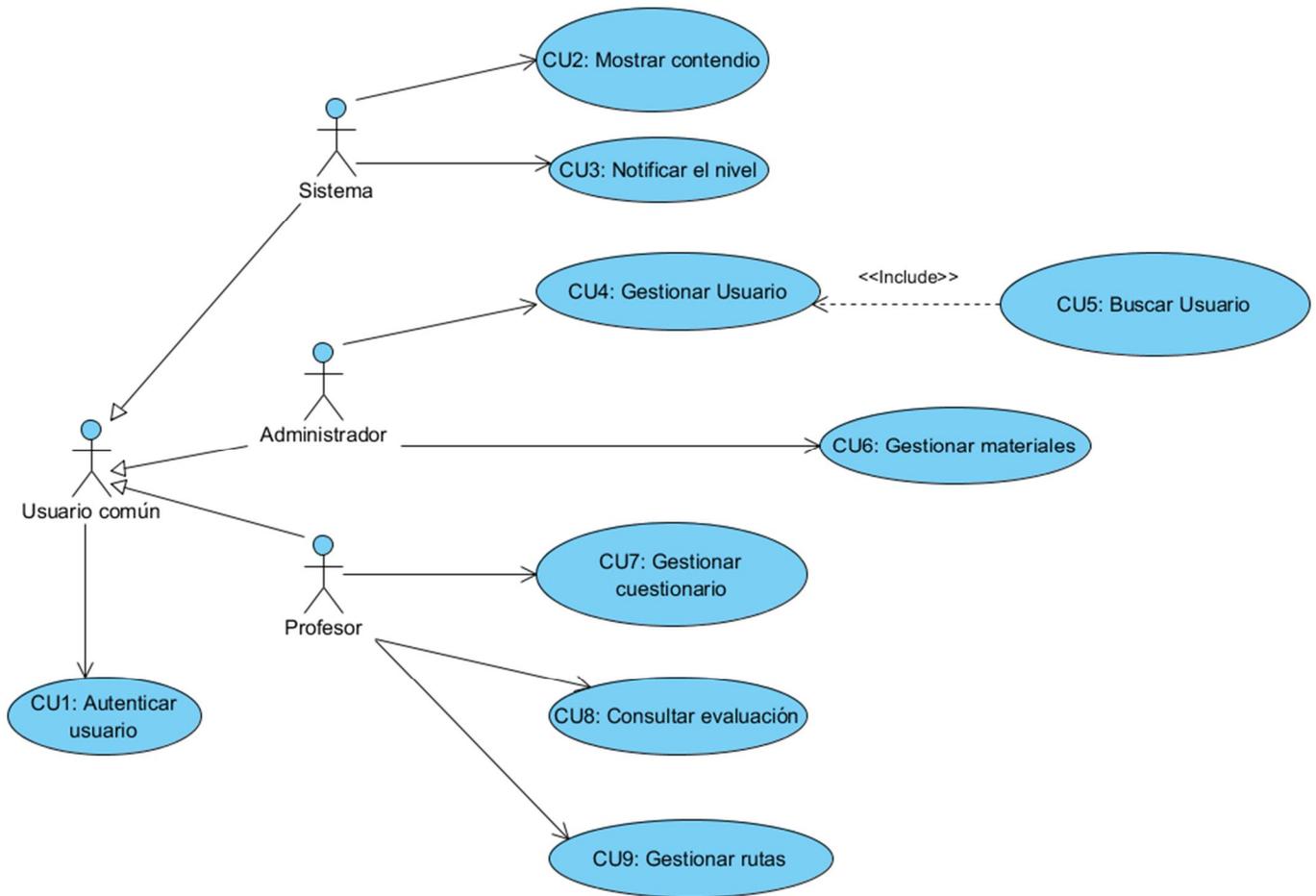


Figura 7: Casos de uso del sistema.

## 2.8- Descripción de los casos de uso del sistema.

En las siguientes tablas, se realiza una descripción de los casos de uso identificados anteriormente y los demás pueden ser consultados en el anexo 2.

Tabla 11: CU1: Autenticar Usuario.

| CU1   | Autenticar usuario               |
|-------|----------------------------------|
| Actor | Sistema, Administrador, Profesor |

|                    |  |
|--------------------|--|
| <b>Descripción</b> | Es el caso de uso común a los 3 actores, pues los profesores y administradores necesitan autenticarse para tener acceso al sistema, y para que el sistema ejecute sus casos de uso, los usuarios deben estar autenticados. |
|--------------------|--|

Tabla 12: CU2: Mostrar contenido.

|                    |  |
|--------------------|--|
| <b>CU2</b>         | <b>Mostrar contenido</b>   |
| <b>Actor</b>       | Sistema  |
| <b>Descripción</b> | Este caso de uso tiene la función de permitir al sistema mostrar los contenidos relacionados con la asignatura que se va a impartir. |

Tabla 13: CU3: Notificar el nivel

|                    |  |
|--------------------|--|
| <b>CU3</b>         | <b>Notificar el nivel</b>  |
| <b>Actor</b>       | Sistema  |
| <b>Descripción</b> | El sistema debe notificar el nivel de aprendizaje de los estudiantes después que el profesor realice el cuestionario para esto los estudiantes deberán llenar los campos correspondientes a las preguntas que se realizarán en él. |

Tabla 14: CU4: Gestionar Usuario

|                    |   |
|--------------------|---|
| <b>CU4</b>         | <b>Gestionar Usuario</b>  |
| <b>Actor</b>       | Administrador   |
| <b>Descripción</b> | Este caso de uso, permite al administrador del sistema gestionar todas las actividades realizadas por el usuario, puede insertar , eliminar o modificar los datos del usuario |

Tabla 15: CU5: Buscar Usuario.

|                    |   |
|--------------------|---|
| <b>CU5</b>         | <b>Buscar Usuario</b>   |
| <b>Actor</b>       | Administrador   |
| <b>Descripción</b> | Ese caso de uso permite al sistema buscar los usuarios registrados. |

Tabla 16: CU6: Gestionar materiales.

|                    |   |
|--------------------|---|
| <b>CU6</b>         | <b>Gestionar materiales</b>   |
| <b>Actor</b>       | Administrador   |
| <b>Descripción</b> | Permite al administrador del sistema gestionar los materiales de aprendizaje de acuerdo con el tamaño de los mismos, las opciones que tiene el mismo son: eliminar, modificar, y actualizar materiales. |

Tabla 17: CU7: Gestionar cuestionario.

|                    |   |
|--------------------|---|
| <b>CU7</b>         | <b>Gestionar cuestionario</b>   |
| <b>Actor</b>       | Profesor  |
| <b>Descripción</b> | Permite al administrador del sistema gestionar los cuestionarios con las preguntas después que el usuario se autentica en el sistema las opciones que tiene el mismo son: crear, eliminar, modificar, y actualizar el cuestionario. |

Tabla 18: CU8: Consultar evaluación.

|                    |   |
|--------------------|---|
| <b>CU8</b>         | <b>Consultar evaluación</b>   |
| <b>Actor</b>       | Profesor  |
| <b>Descripción</b> | Permite al profesor consultar las evaluaciones de las preguntas de las tareas realizadas. |

Tabla 19:CU9: Gestionar rutas.

| CU9         | Gestionar rutas  |
|-------------|--|
| Actor       | Profesor   |
| Descripción | Permite al profesor gestionar y establecer las estrategias pedagógicas después que el usuario se autentica en el sistema y responde el cuestionario las opciones que tiene el mismo son: crear, eliminar, modificar, y actualizar el cuestionario. |

## 2.9- Diagrama de Clases

Un diagrama de clases en Lenguaje Unificado de Modelado (*UML*) es un tipo de diagrama de estructura estática que describe la estructura de un sistema mostrando las clases del sistema, sus atributos, operaciones (o métodos), y las relaciones entre los objetos( (Bedoya, 2017).

Los diagramas de clases contienen normalmente los siguientes elementos, clases, interfaces, colaboraciones, relaciones de dependencia, generalización y asociación. Estos se utilizan para modelar la vista estática de un sistema, muestra principalmente los requisitos funcionales de un sistema y los servicios que el sistema debe proporcionar a sus usuarios finales.

Las clases se representan por rectángulos que muestran el nombre de la clase y opcionalmente el nombre de las operaciones y atributos. Los compartimientos se usan para dividir el nombre de la clase, atributos y operaciones. Adicionalmente las restricciones, valores iniciales y parámetros se pueden asignar a clases

### 2.9.1- Diagrama de Clases de diseño

Un Diagrama de clases de diseño muestra la especificación para las clases de software de una aplicación. Incluye la siguiente información:

- Clases, asociaciones y atributos.
- Interfaces, con sus operaciones y constantes.

- Métodos.
- Navegabilidad.
- Dependencias

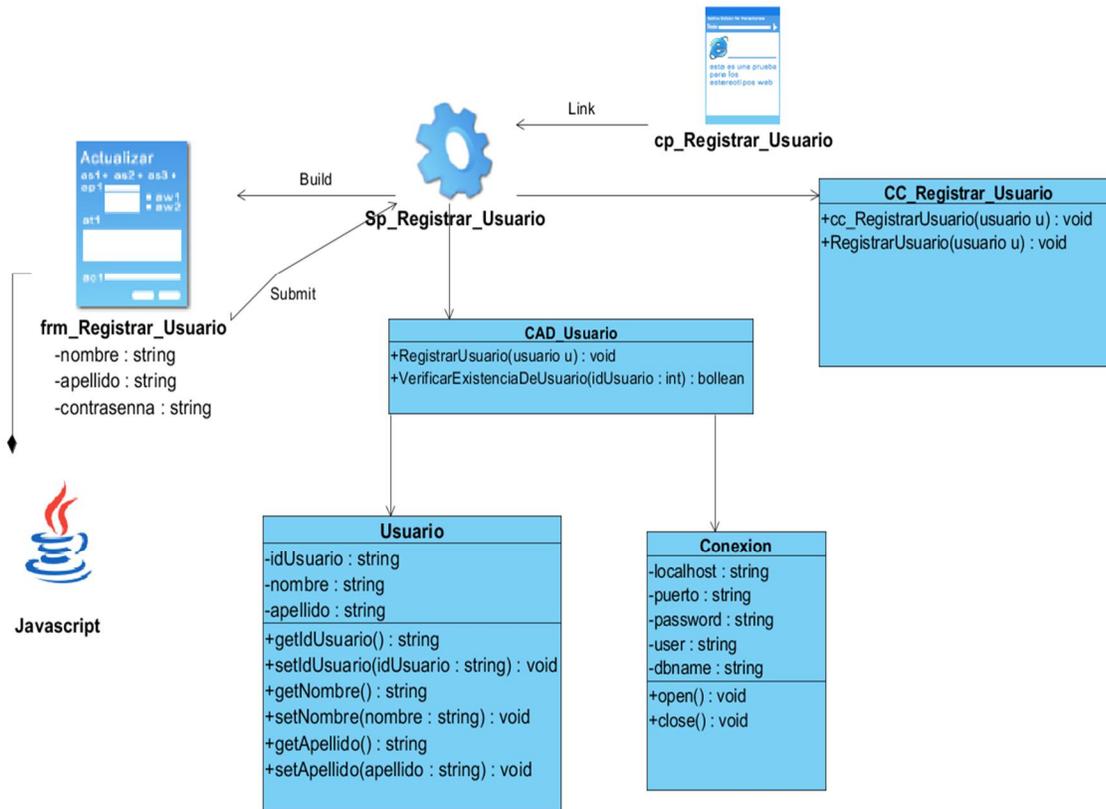


Figura 8: Diagrama de clases Registrar Usuario

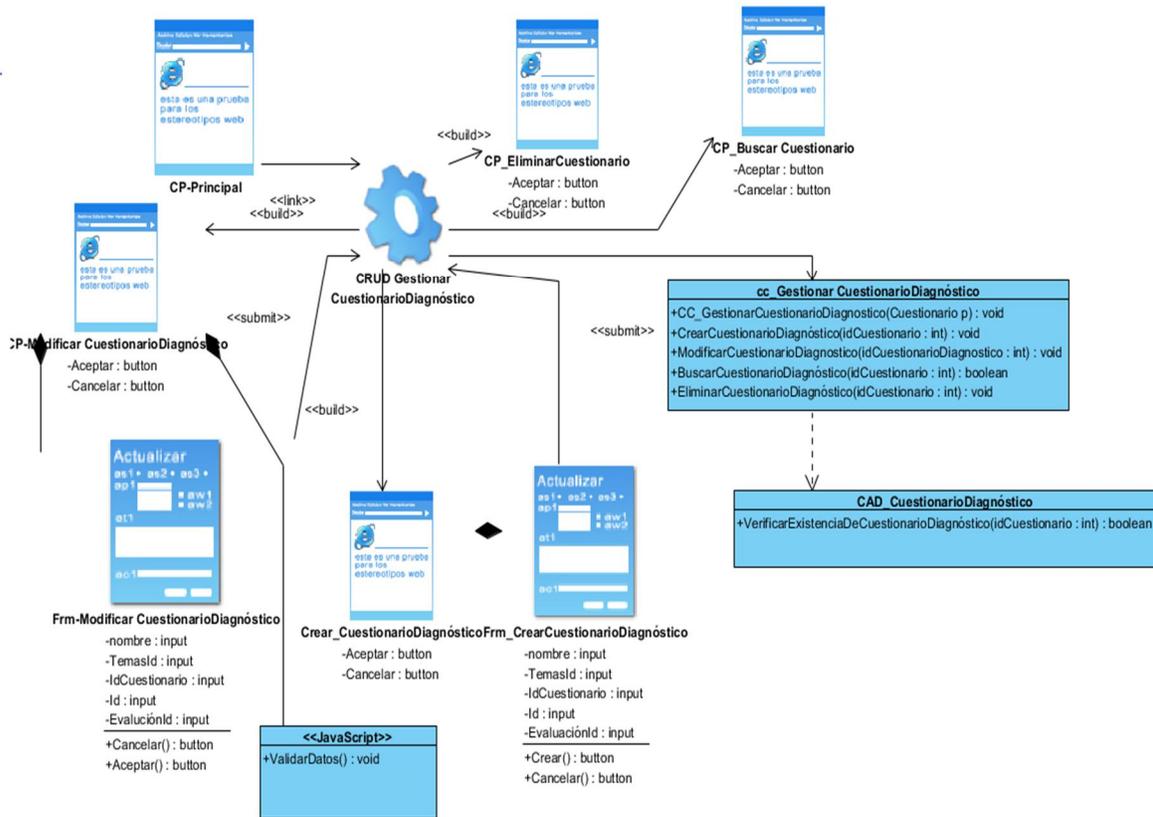


Figura 9: Diagrama de clase de diseño Gestionar cuestionario (elaboración propia).

## 2.10- Modelo de datos

El modelo Entidad-Relación facilita al ingeniero del software identificar los objetos de datos y sus relaciones por medio de una notación gráfica, define en su análisis estructural los datos que son adicionados, almacenados, modificados y producidos dentro de una aplicación (Pressman, 2011).

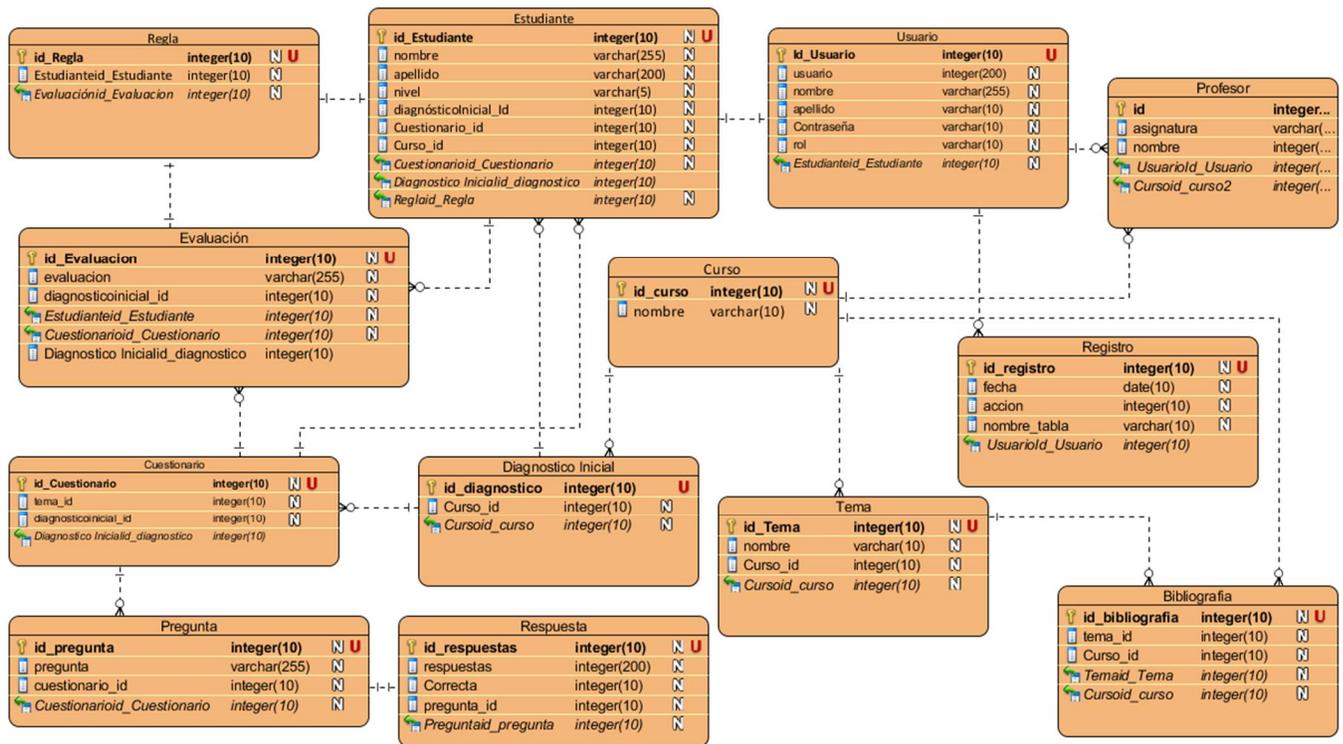


Figura 10: Modelo de datos (elaboración propia).

## 2.11- Modelo de Despliegue

El diagrama de despliegue es un tipo de diagrama del Lenguaje Unificado de Modelado que se utiliza para modelar la disposición física de los artefactos de software en nodos (usualmente plataforma de hardware). Muestra la arquitectura del sistema como el despliegue (la distribución) de los artefactos de software a los objetivos de despliegue (Pressman, 2010).

Un nodo es un elemento físico que existe en tiempo de ejecución y representa un recurso computacional, que existe en tiempo de ejecución y representa un recurso computacional, que generalmente tiene memoria y a menudo, capacidad de procesamiento. Los nodos se utilizan para modelar la topología del hardware sobre el que se ejecuta el sistema. Representan un procesador o un dispositivo sobre el que se pueden desplegar los componentes. La relación entre nodo y componente que despliega puede mostrarse como una relación de dependencia (Ruiz, 2008).

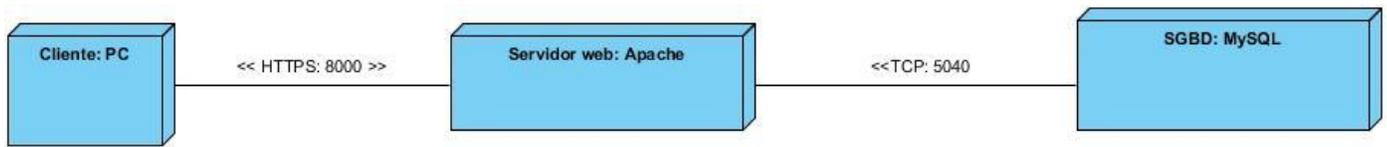


Figura 11: Diagrama de despliegue (elaboración propia).

**PC Cliente:** Se refiere a las estaciones de trabajo que realizan las peticiones al servidor de aplicaciones donde está hospedada la plataforma que soporta los módulos tutor y dominio para el proceso de enseñanza-aprendizaje de la inteligencia artificial, mediante un navegador web utilizando el protocolo de comunicación *HTTPS* por el puerto 8000.

**Servidor Web:** Es el encargado de brindar la interfaz de la plataforma para que los usuarios puedan hacer uso de esta, almacena todo el código fuente del sistema y se comunica por medio de los protocolos *TCP* con el servidor de bases de datos.

**Sistema Gestor de Bases de Datos:** Almacena toda la información que brinda la plataforma hospedada en el servidor de aplicaciones. La información es obtenida o modificada en dependencia del nivel de privilegio del usuario que realiza la petición. La comunicación con el servidor de aplicaciones es a través del protocolo *TCP* empleando el puerto 5040.

## 2.12- Patrón Arquitectónico

Los patrones de la arquitectura de un sistema, definen la estructura fundamental con la que se va a desarrollar el software. Para el desarrollo del STI el patrón que se utilizará es el Modelo-Vista - Controlador (MVC).

El *framework* de desarrollo web Django emplea una modificación del estilo arquitectónico Modelo-Vista Controlador (MVC), llamada Model-Template-View (MTV), que sería Modelo-Plantilla-Vista, esta forma de trabajar permite que sea pragmático. Como plantea Infante-Montero (2012), para comprender cómo funciona el MTV de Django se debe tener en cuenta, su analogía con el MVC de la siguiente manera:

- El modelo en Django continúa siendo Modelo.
- La vista en Django pasa a llamarse Plantilla.

- El controlador en Django pasa a llamarse Vista.

A partir de lo planteado se asumirá el estilo arquitectónico MTV de Django del cual se explica su funcionamiento a continuación y se podrá observar en la figura 12:

- El navegador web envía una solicitud
- La vista interactúa con el modelo para obtener los datos
- La vista llama a la plantilla
- La plantilla muestra la respuesta a la solicitud del navegador

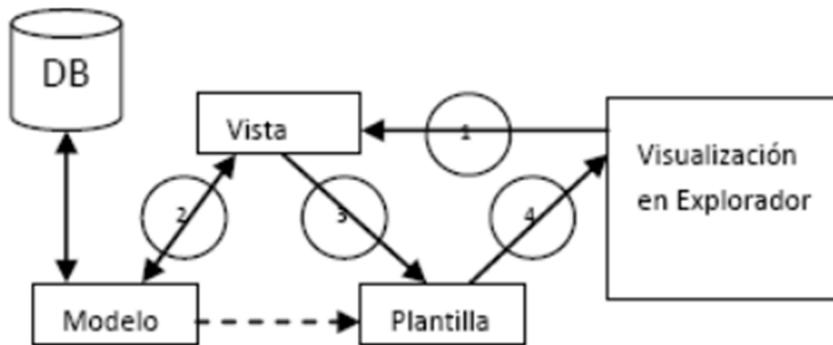


Figura 12: Esquema simplificado de la arquitectura MVP de Django (Condori, 2013).

## 2.13- Patrones de Diseño

Un patrón es una descripción del problema y la esencia de su solución, de modo que la solución puede reutilizarse en diferentes configuraciones. El patrón no es una especificación detallada. Más bien, puede considerarla como una descripción de sabiduría y experiencia acumuladas, una solución bien probada a un problema común (Sommerville, 2011).

### 2.13.1- Patrones GRASP

Los Patrones de Principios Generales para Asignar Responsabilidades (*GRASP* por sus siglas en inglés) describen los principios fundamentales del diseño de objetos y la asignación de responsabilidades, expresados como patrones. Experto es un patrón que se usa más que cualquier otro al asignar

responsabilidades; es un principio básico que suele utilizarse en el diseño orientado a objetos. Dentro de los patrones GRASP utilizados para el desarrollo del STI, se encuentran los siguientes:

**Experto:** Las responsabilidades deben ser asignadas a las clases que tiene la información necesaria para cumplirla. Esto se evidencia en las clases modelos y las clases controladoras. Por ejemplo, la clase controladora cuestionario, será la encargada de insertar, modificar y eliminar un cuestionario (gomez, 2014).

**Creador:** Con la utilización de este patrón se determina cuáles serán clases responsables de crear una nueva instancia de la clase que realiza la conexión a la base de datos. Una de las consecuencias de usar este patrón es la visibilidad entre la clase creada y la clase creadora. Una ventaja es el bajo acoplamiento, lo cual supone facilidad de mantenimiento y reutilización (rodriguez, 2016).

**Controlador:** establece una clara separación entre la interfaz de usuario (una interfaz gráfica, por ejemplo) y el corazón o núcleo de procesamiento de la aplicación, donde se halla la lógica del negocio. La idea básica es crear una clase que implemente métodos dedicados a "escuchar" o "atender" los eventos del sistema. Un ejemplo de esto se evidencia en la clase de inicio de un sistema, ya que es la controladora que se encarga de mostrar la página de inicio del sistema (Bermudez, 2013).

**Bajo Acoplamiento:** Cuando se habla de acoplamiento entre objetos, se hace referencia a la "fuerza" con la que ciertos objetos están relacionados, o dependen unos de otros. Mientras más dependencias tengan un objeto de otros para llevar a cabo sus tareas, más fuerte será el acoplamiento. Por tanto este patrón es el encargado de asignar una responsabilidad para mantener bajo acoplamiento con el objetivo de tener las clases lo menos ligadas entre sí, de forma que si se realiza modificación en alguna de ellas, tenga la mínima repercusión en el resto de las clases. Esto se evidencia en todas las clases ya que un cambio en las clases modelo no implica cambios en las controladoras ni en las vistas (susana, 2005).

**Alta Cohesión:** es el encargado de asignar una responsabilidad de modo que la cohesión siga siendo alta, con el objetivo de que cada una de las clases tenga responsabilidades específicas, por lo que cada clase tiene determinadas responsabilidades y colabora con otras para llevarlas a cabo, como por ejemplo las clases controladoras modifican los datos haciendo uso de las clases modelos, las cuales solo tienen la responsabilidad de hacer cambios en la base de datos (Escalante, 2016).

### 2.13.2- Patrones GOF

Representan patrones que dan soluciones técnicas basadas en programación orientada a objetos (POO) que favorecen la reutilización del código (Mühlrad, 2008).

**Decorador:** es un patrón estructural que añade dinámicamente nuevas responsabilidades a un objeto, proporcionando una alternativa flexible a la herencia para extender la funcionalidad. El patrón Decorador responde a la necesidad de añadir dinámicamente funcionalidad a un objeto esto nos permite no tener que crear sucesivas clases que hereden de la primera incorporando la nueva funcionalidad, sino otras que la implementan y se asocian a la primera. Provee una alternativa muy flexible para agregar funcionalidad a una clase. Como ejemplo de este patrón se evidenciará la utilización de los decoradores de python o definidos por el programador.

### 2.14- Diagrama de Componentes

El diagrama de componentes muestra las relaciones estructurales entre los componentes de un sistema. Los componentes se consideran unidades autónomas encapsuladas dentro de un sistema o subsistema que proporcionan una o más interfaces. Estos diagramas son generalmente dirigidos al personal de aplicación de un sistema, además presenta una comprensión temprana del sistema global que se está construyendo (Bell, 2004).

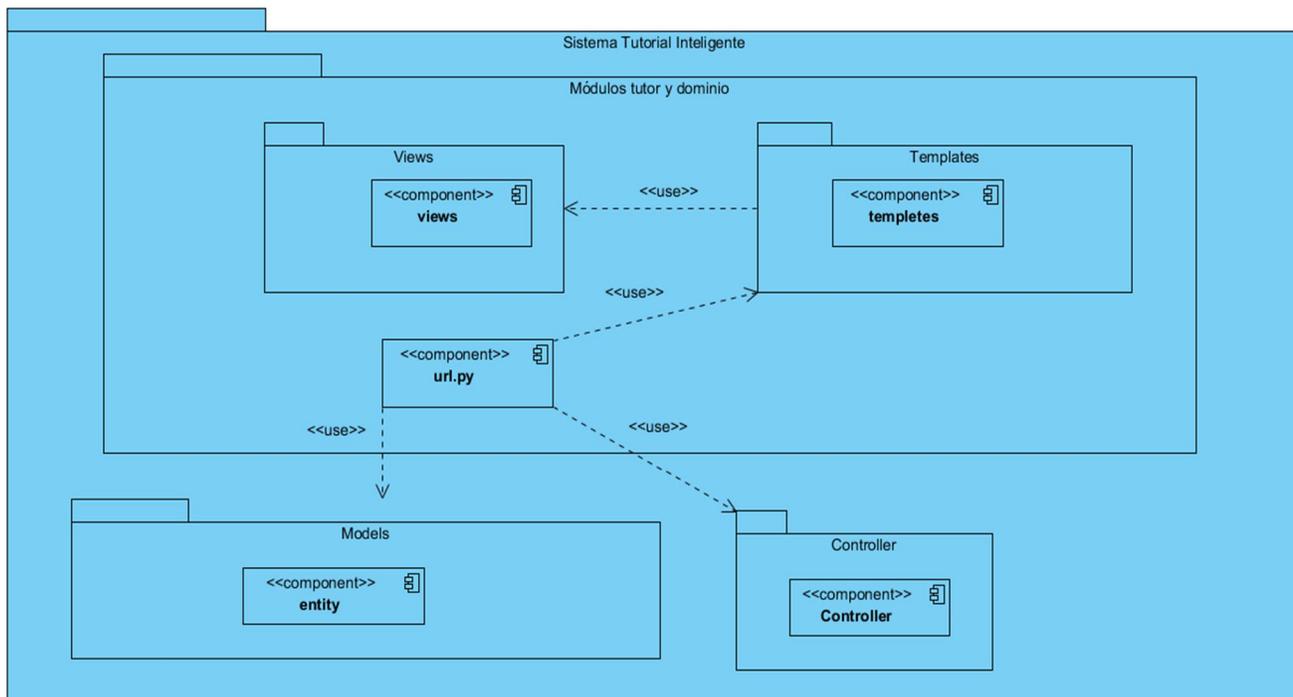


Figura 13: Diagrama de componentes (elaboración propia).

## 2.15- Conclusiones del Capítulo

A partir del análisis desarrollado en este capítulo, se identificaron varios requisitos funcionales de suma importancia para el cumplimiento del desarrollo de los módulos del sistema, y se describió la propuesta de solución. Se determinaron los requerimientos del sistema para el posterior desarrollo de los módulos tutor y dominio perteneciente al STI para dar cumplimiento de la propuesta de solución. Basándose en la metodología de desarrollo AUP con escenario 2, se identificaron las fases de la metodología inicio, ejecución y cierre, en la que se confeccionó los requisitos funcionales necesarios para el sistema y los diagramas de casos de uso referentes a los requerimientos determinado como parte de la propuesta de solución, en la que se especificaron los datos necesarios para la futura implementación del sistema.

## Conclusiones generales

En la presente investigación se realiza el análisis y diseño de los módulos tutor y dominio de un sistema tutorial inteligente (STI). Una vez completada la investigación, se puede concluir lo siguiente:

- Se reflejaron todos los requisitos a tener en cuenta para el análisis y diseño para la futura implementación de los módulos tutor y dominio de un STI, capaz de ayudar a los estudiantes de la asignatura Inteligencia Artificial 1, a consolidar su conocimiento y lograr aprovechar su tiempo de auto estudio, mostrándoles la bibliografía más acorde a sus características individuales.
- Se determinó cómo están estructurados los STI y las diferentes técnicas de IA que se utilizan para el desarrollo de estas herramientas educativas. Lo que permitió la selección adecuada de la técnica de IA a utilizar, así como los elementos a tener en cuenta para la propuesta de solución.
- La obtención de los artefactos generados por la metodología AUP en el desarrollo de software permitió puntualizar los requisitos funcionales y no funcionales que deberá cumplir el sistema, para dar respuesta a las necesidades planteadas en la asignatura Inteligencia Artificial 1.
- La investigación llevada a cabo ha permitido reunir los aspectos más relevantes del análisis y diseño de los módulos tutor y dominio de un STI como una herramienta óptima para lograr un proceso de enseñanza-aprendizaje. Se ha destacado la ventaja de los tutores inteligentes en cuanto se adaptan a las características del estudiante y de acuerdo a estas diseña la mejor estrategia pedagógica para dirigirse a él y lograr su aprendizaje. Los conceptos plasmados en este documento han sido producto de un extenso proceso de investigación, en el cual se ha descubierto no solo el impacto de los STI sino también su importancia como apoyo al proceso de aprendizaje, y la forma como este aplica diferentes técnicas y métodos de la IA para lograr los objetivos definidos.

## Recomendaciones

Una vez concluida la investigación, para la futura implantación del sistema el autor recomienda:

- La implementación del sistema constituido por los módulos tutor y dominio de un sistema tutorial inteligente como apoyo al proceso de enseñanza aprendizaje de la inteligencia artificial 1 en la universidad de la ciencias informáticas.
- Extender alcance del sistema, que no sea solo para la asignatura inteligencia artificial 1, pero también para todas las demás asignaturas que componen el plan de estudio de la carrera en la universidad de las Ciencias Informáticas.
- Para la implementación futura desarrollar funcionalidades estadísticas que permita controlar la evolución del estudiante.

## Referencias Bibliográficas

- Alvarado Aldo Ramirez. 2017.** *Introducción al Machine Learning*. Universidad Mayor de San Andres : s.n., 2017.
- Anita Ferreira Cabrera. 2007.** *Sistema tutorial inteligente para el tratamiento de los errores gramaticales para el español como Lengua Extranjera con fines académicos*. Universidad de Concepción : s.n., 2007.
- Bedoya, Chávez. 2017.** *Diagramas de Clase UML*. 2017.
- Bell. 2004.** *Diagrama de Componentes*. 2004.
- Bermudez, Camilo. 2013.** *Patrones de Diseño*. 2013.
- Builes, Jovani Alberto Jiménez. 2012.** *Trabajo de Investigación Sobre Ingeniería de Sistemas*. 2012.
- Castillo. 1998.** *sistemas inteligentes para el modelado del tutor*. 1998.
- Cataldi. 2010.** *Los sistemas tutores inteligentes*. 2010.
- . *Sistemas Tutores Inteligentes para el Modelado del Tutor*.
- . **2009.** *sistemas tutores inteligentes orientados a la enseñanza*. Buenos Aires : s.n., 2009.
- . **2009.** *SISTEMAS TUTORES INTELIGENTES ORIENTADOS A LA ENSEÑANZA*. Buenos Aires : EDUTEC, 2009. 1135-9250.
- . **2009.** *Sistemas tutores inteligentes, procedimientos y metodos*. 2009.
- . **2010.** *Sistemas tutoriales inteligentes para la enseñanza*. 2010.
- Chávez, Bedoya. 2017.** *Diagramas de Clase UML*. 2017.
- Condori. 2013.** *Programando en Python*. 2013.
- Davis Falkenauer. 2001.** *Sistemas tutores inteligentes orientados a la enseñanza*. 2001.
- Davis, Falkenauer. 1999.** *sistemas tutores inteligentes orientados a enseñanza*. 1999.
- EcuRed.** *requisitos no funcionales*.
- Escalante. 2016.** *Patrones de Diseño*. 2016.
- Fernando, Salgueiro. 2010.** *Sistemas Tutores Inteligentes para el Modelado del Tutor*. 2010.
- Ferrer, Lourdes Alicia Risco. 2017.** *Sistema Tutorial Inteligente para la asignatura Reconocimiento de Patrones*. 2017.

- García, Alejandro Alfonso Pérez. 2008.** *Desarrollo de herramientas web de gestión docente.* 2008.
- Gardner. 2001.** *Teoría de Las Inteligencias Múltiples.* 2001.
- Gauchat. 2014.** *Lenguajes marcado de hipertexto.* 2014.
- Giraffa. 1997.** *sistemas tutores inteligentes orientados a la enseñanza.* 1997.
- gomez, javier. 2014.** *Patrones de Diseño de software.* 2014.
- Hernández, Jairo A. Durango. 2015.** *Los Sistemas Tutores Inteligentes y su Aplicabilidad en la Educación.* Colombia : s.n., 2015.
- Inteligencia Artificial. 2018.** *Inteligencia Artificial.* 2018.
- José Manuel Gutiérrez. 2016.** *Sistemas Expertos Basados en Reglas.* Dpto. de Matemática Aplicada. Universidad de Cantabria : s.n., 2016.
- Lage, Zulma y. 2009.** *sistemas tutores inteligentes orientados a la enseñanza.* 2009.
- Language de Programación. 2019.** *Definiciones de Lenguajes de Programación.* 2019.
- Luisa Saavedra. 2016.** *PROTOTIPO DE TUTOR INTELIGENTE PARA EL APRENDIZAJE DE LA PROGRAMACIÓN DE COMPUTADORES.* Colombia : s.n., 2016.
- Luisa Idorka Morales . 2012.** *Sistema Tutorial Inteligente de apoyo al proceso de enseñanza-aprendizaje de la asignatura sistemas operativos.* Habana : s.n., 2012.
- (JetBrains. 2016.** *Entorno de desarrollo pycharm.* 2016.
- .O, Rodriguez. 2010.** *Entorno Integrado de desarrollo.* 2010.
- Machine Learning. 2018.** *Introducción a la Inteligencia Artificial.* 2018.
- Maribel García. 2012.** *Sistema experto para el Diagnóstico y Tratamiento de Infecciones de Transmisión Sexual.* Universidad de Cienfuegos , Cuba : s.n., 2012.
- Mühlrad, Daniel. 2008.** *Patrones de diseño.* 2008.
- MySQL. 2018.** MySQL. [En línea] 2018. <http://www.mysql.com>.
- Natalia Martínez Sánchez. 2011.** *diseño de sistemas de enseñanza-aprendizaje inteligentes utilizando el razonamiento basado en casos.* s.l. : Revista Generación Digital., 2011.
- . **2011.** *Sistemas de enseñanza - aprendizaje inteligentes y los sistemas basados en casos.* 2011.
- Pressman, Roger. 2011.** *Ingeniería de software.* 2011.
- . **2010.** *Ingeniería de Software.* 2010.

- . 2005. *Ingeniería del Software*. 2005.
- rodriguez. 2016. *Patrones de diseño*. 2016.
- Rodriguez., O. 2010. 2010.
- Rossun, Guido Van. 2009. *El tutorial de Python*. 2009.
- Ruiz, Francisco. 2008. *INGENIERÍA DEL SOFTWARE I*. 2008.
- Salgueiro, Fernando. 2010. *Sistemas Tutores Inteligentes*. 2010.
- Salgueiro, Fernando. 2005. *Sistemas inteligentes para el modelado del tutor*. 2005.
- . 2010. *Sistemas Tutores Inteligentes para el Modelado del Tutor*. 2010.
- . 2011. *Sistemas Tutores Inteligentes Para el Modelado del Tutor*. 2011.
- Sevilla, Fernández, y Díaz. 2017. *Lenguaje de programación python*. 2017.
- SISTEMAS TUTORES INTELIGENTES ORIENTADOS A LA ENSEÑANZA PARA LA COMPRESIÓN*. Zulma Cataldi, Fernando J.Lage. 2009. 28, Buenos Aires : Numero 28, 2009. 1135-9250.
- Somerville. 2010. *technology y Engineering*. 2010.
- Sommerville. 2010. *Ingeniería de software*. 2010.
- . 2011. *Ingeniería de Software*. 2011.
- . 2006. *Ingeniería de Software*. 2006.
- . 2011. *Modelo Conceptual*. 2011.
- susana, Montero. 2005. *Patrones de Diseño*. 2005.
- Tamara Rodríguez Sánchez. 2015. *Metodología de desarrollo para la actividad productiva de la UCI*. Habana : s.n., 2015.
- UML, Lenguaje. 2018. *Lenguaje unificado del modelado*. 2018.
- VANLEHN. 1998. *sistemas tutores inteligentes orientados a la enseñanza*. Buenos Aires : Facultad Regional Buenos Aires Universidad Tecnológica Nacional., 1998. 1135-9250.
- Wenger. 1997. *sistemas tutores inteligentes orientados a la enseñanza*. 1997.
- phpMyAdmin. 2016. *phpMyAdmin*. 2016.

**Wolf. 1984.** *SISTEMAS TUTORES INTELIGENTES ORIENTADOS A LA enseñanza.* Buenos Aires : s.n., 1984. 1135-9250.

**Zulma Cataldi. 2009.** *Articulos sobre la Arquitectura de un Sistem Tutorial Inteligente.* España : s.n., 2009.

## Bibliografía

**Aguilar, R. M. R., González, J. L. M. C., y Campos, A. L. L. (2013).** Diseño de un sistema tutorial inteligente. *Apertura*, 5(1), 36-47.

**Alshawwa, Izzeddin A., Mohammed O. Al-Shawwa, y Samy S. Abu-Naser. 2019.** «An Intelligent Tutoring System for Learning Computer Network CCNA». <http://dspace.alazhar.edu.ps/xmlui/handle/123456789/129>.

**Alvarado Aldo Ramirez. 2017.** *Introducción al Machine Learning*. Universidad Mayor de San Andres : s.n., 2017.

**Anita Ferreira Cabrera. 2007.** *Sistema tutorial inteligente para el tratamiento de los errores gramaticales para el español como Lengua Extranjera con fines académicos*. Universidad de Concepción : s.n., 2007.

**Annabel Latham.** *A conversational intelligent tutoring system to automatically predict learning styles* The Intelligent Systems Group, School of Computing, Mathematics and Digital Technology.

**Bedoya, Chávez. 2017.** *Diagramas de Clase UML*. 2017.

**Bell. 2004.** *Diagrama de Componentes*. 2004.

**Bermudez, Camilo. 2013.** *Patrones de Diseño*. 2013.

**Builes, Jovani Alberto Jiménez. 2012.** *Trabajo de Investigación Sobre Ingeniería de Sistemas*. 2012.

**Caro, Manuel F., Darsana P. Josyula, y Jovani A. Jiménez. 2015.** «Multi-Level Pedagogical Model for the Personalization of Pedagogical Strategies in Intelligent Tutoring Systems». *Dyna* 82 (194): 185-93.

**Castillo. 1998.** *sistemas inteligentes para el modelado del tutor*. 1998.

**Castrillón, E. P. (2004).** Sistemas tutoriales inteligentes, un aporte de la inteligencia artificial para la mediación pedagógica. *Revista Virtual Universidad Católica del Norte*, 1(12). Recuperado de <https://revistavirtual.ucn.edu.co/index.php/RevistaUCN/article/view/279>

**Cataldi. 2010.** *Los sistemas tutores inteligentes*. 2010.

—. *Sistemas Tutores Inteligentes para el Modelado del Tutor*.

**Cataldi, Zulma. 2010.** *Los sistemas Tutores Inteligentes*. 2010.

—. **2009.** *sistemas tutores inteligentes orientados a la enseñanza*. Buenos Aires : s.n., 2009

—. **2009.** *sistemas tutores inteligentes orientados para la enseñanza*. Buenos Aires : Facultad Regional Buenos Aires Universidad Tecnológica Nacional., 2009. 1135-9250.

—. **2009.** *Sistemas tutores inteligentes, procedimientos y metodos*. 2009.

—. **2010.** *Sistemas tutoriales inteligentes para la enseñanza*. 2010.

- Chávez, Bedoya. 2017.** *Diagramas de Clase UML.* 2017.
- (Conati, Cristina, 2010).** *Intelligent Tutoring Systems: New Challenges and Directions.*
- Condori. 2013.** *Programando en Python.* 2013.
- Davis Falkenauer. 2001.** *Sistemas tutores inteligentes orientados a la enseñanza.* 2001.
- Davis, Falkenauer. 1999.** *sistemas tutores inteligentes orientados a enseñanza.* 1999.
- Eagle, Michael, y Tiffany Barnes. 2010.** «Intelligent Tutoring Systems, Educational Data Mining, and the Design and Evaluation of Video Games». En *Intelligent Tutoring Systems*, editado por Vincent Aleven, Judy Kay, y Jack Mostow, 215-17. *Lecture Notes in Computer Science.* Springer Berlin Heidelberg.
- EcuRed. requisitos no funcionales.**
- 2016.** *Ingeniería de software 2.* 2016.
- Escalante. 2016.** *Patrones de Diseño.* 2016.
- Fernando, Salgueiro. 2010.** *Sistemas Tutores Inteligentes para el Modelado del Tutor.* 2010.
- Ferrer, Lourdes Alicia Risco. 2017.** *Sistema Tutorial Inteligente para la asignatura Reconocimiento de Patrones.* 2017.
- García, Alejandro Alfonso Pérez. 2008.** *Desarrollo de herramientas web de gestión docente.* 2008.
- Gardner. 2001.** *Teoría de Las Inteligencias Múltiples.* 2001.
- Gauchat. 2014.** *Lenguajes marcado de hipertexto.* 2014.
- Giraffa. 1997.** *sistemas tutores inteligentes orientados a la enseñanza.* 1997.
- gomez, javier. 2014.** *Patrones de Diseño de software.* 2014.
- Gordillo-Guillén, A., Andrade, H., y Rivera-Lopez, R. (2017).** Modelo de un sistema tutor inteligente para el desarrollo del pensamiento computacional. *Journal CIM 2017*, 5.
- Hernández, Jairo A. Durango. 2015.** *Los Sistemas Tutores Inteligentes y su Aplicabilidad en la Educación.* Colombia : s.n., 2015.
- Huapaya, C. R., Arona, G. M., y Lizarralde, F. A. (2005).** Enseñanza de la Ingeniería con Sistemas Tutoriales Inteligentes. *Información tecnológica*, 16(5), 75-78. <https://doi.org/10.4067/S0718-07642005000500012>
- Inteligencia Artificial. 2018.** *Inteligencia Artificial.* 2018.
- José Manuel Gutiérrez. 2016.** *Sistemas Expertos Basados en Reglas.* Dpto. de Matemática Aplicada. Universidad de Cantabria : s.n., 2016.

- Lage, Zulma y. 2009.** *sistemas tutores inteligentes orientados a la enseñanza.* 2009.
- Lenguaje de Programación. 2019.** *Definiciones de Lenguajes de Programación.* 2019.
- Lourdes Ferrer. 2017.** *Sistema Tutorial Inteligente para la asignatura Reconocimiento de Patrones.* 2017.
- Los sistemas tutoriales inteligentes* Available from: <http://www.redalyc.org/articulo.ao>
- Luisa Saavedra. 2016.** *PROTOTIPO DE TUTOR INTELIGENTE PARA EL APRENDIZAJE DE LA PROGRAMACIÓN DE COMPUTADORES.* Colombia : s.n., 2016.
- Luisa Idorka Morales . 2012.** *Sistema Tutorial Inteligente de apoyo al proceso de enseñanza-aprendizaje de la asignatura sistemas operativos.* Habana : s.n., 2012.
- (JetBrains. 2016.** *Entorno de desarrollo pycharm.* 2016.
- James C. Lester, Rosa Maria Vicari.** *Intelligent Tutoring Systems. 7th International Conference, ITS 2004 Maceió, Alagoas, Brazil, August 30 – September 3, 2004 Proceedings.*
- .O, Rodriguez. 2010.** *Entorno Integrado de desarrollo.* 2010.
- Machine Learning. 2018.** *Introducción a la Inteligencia Artificial.* 2018.
- Mahdi, Ali, Mohammed Alhabbash, y Samy Abu-Naser. 2016.** «An intelligent tutoring system for teaching advanced topics in information security». *World Wide Journal of Multidisciplinary Research and Development* 2 (diciembre): 1-9.
- (Manuel Valenzuela-Rendón, 2016).** *A Tutorial on Deep Neural Networks for Intelligent Systems*
- Modelado de un Sistema tutorial inteligente available from:* [www.scielo.cl/scielo.org](http://www.scielo.cl/scielo.org)
- <http://revistas.udistrital.edu.co/ojs/index.php/revcie/index>
- Maribel García. 2012.** *Sistema experto para el Diagnóstico y Tratamiento de Infecciones de Transmisión Sexual.* Universidad de Cienfuegos , Cuba : s.n., 2012.
- Melo, Francisco S., Samuel Mascarenhas, y Ana Paiva. 2018.** «A Tutorial on Machine Learning for Interactive Pedagogical Systems». *International Journal of Serious Games* 5 (3). <https://doi.org/10.17083/ijsg.v5i3.256>.
- Mühlrad, Daniel. 2008.** *Patrones de diseño.* 2008.
- MySQL. 2018.** MySQL. [En línea] 2018. <http://www.mysql.com>.
- Natalia Martínez Sánchez. 2011.** *diseño de sistemas de enseñanza-aprendizaje inteligentes utilizando el razonamiento basado en casos.* s.l. : Revista Generación Digital., 2011.
- . **2011.** *Sistemas de enseñanza - aprendizaje y los sistemas basados en casos.* 2011.

**Pamela J. Woods and James R. Warren.2010.** *Rapid Prototyping of an Intelligent Tutorial System. Advanced Computing Research Centre University of South Australia pam.woods, [james.warren@unisa.edu.au](mailto:james.warren@unisa.edu.au).*

**Pressman. 2011.** *Ingenieria del Software.* 2011.

—. **2010.** *Ingenieria de Software.* 2010.

**Pressman, Rojer. 2005.** *Ingenieria del software.* 2005.

**rodriguez. 2016.** *Patrones de diseño.* 2016.

**Rodriguez., O. 2010.** 2010.

**Rossun, Guido Van. 2009.** *El tutorial de Paython.* 2009.

**Ruiz, Francisco. 2008.** *INGENIERÍA DEL SOFTWARE I.* 2008.

**Salgueiro, Fernado. 2010.** *Sistemas Tutores Inteligentes.* 2010.

**Salgueiro, Fernando A.** *sistemas Inteligentes para el modelado del Tutor.* 2005.

—. **2010.** *Sistemas Tutores Inteligentes para el Modelado del Tutor.* 2010.

—. **2011.** *Sistemas Tutores Inteligentes Para el Modelado del Tutor.* 2011.

**Sevilla, Fernández, y Díaz. 2017.** *Lenguaje de programación python.* 2017.

*SISTEMAS TUTORES INTELIGENTES ORIENTADOS A LA ENSEÑANZA PARA LA COMPRENSIÓN.*

**Zulma Cataldi, Fernando J.Lage. 2009.** 28, Buenos Aires : Numero 28, 2009. 1135-9250.

**Somerville. 2010.** *technology y Engineering.* 2010.

**Sommerville. 2010.** *Ingenieria de software.* 2010.

—. **2011.** *Ingenieria de Software.* 2011.

—. **2006.** *Ingenieria de Software.* 2006.

—. **2011.** *Mapa Conceptual.* 2011.

**susana, Montero. 2005.** *Patrones de Diseño.* 2005.

**Tamara Rodríguez Sánchez, Sánchez. 2015.** *Metodologia de desarrollo para la actividad productiva de la uci.* La Habana, universidad de las ciencias informáticas : s.n., 2015.

**Tom Murray.2007.** *Authoring Intelligent Tutoring Systems: An analysis of the state of the art. International Journal of Artificial Intelligence in Education (IJAIED), 1999, 10, pp.98-129. Ffal-00197339f.*

**UML, Lenguaje. 2018.** *Lenguaje unificado del modelado.* 2018.

**VANLEHN. 1998.** *sistemas tutores inteligentes orientados a la enseñanza.* Buenos Aires : Facultad Regional Buenos Aires Universidad Tecnológica Nacional., 1998. 1135-9250.

**Wenger. 1997.** *sistemas tutores inteligentes orientados a la enseñanza.* 1997.

**phpMyAdmin. 2016.** *phpMyAdmin.* 2016. <https://www.phpmyadmin.net>

**Wolf. 1984.** *SISTEMAS TUTORES INTELIGENTES ORIENTADOS A LA enseñanza.* Buenos Aires : s.n., 1984. 1135-9250.

**Zulma Cataldi. 2009.** *Articulos sobre la Arquitectura de un Sistem Tutorial Inteligente.* España : s.n., 2009.

## **Anexos**

### **Anexo 1: Preguntas de la entrevista realizada a los clientes, que en este caso son los profesores de IA:**

#### **Pregunta 1:**

¿Cuáles son las principales herramientas y metodologías utilizadas para el posterior desarrollo de los módulos tutor y dominio de un sistema tutorial inteligente?

#### **Pregunta 2:**

¿Cuáles son las principales deficiencias y dificultades que presenta los estudiantes de la UCI en la asignatura inteligencia artificial 1?

#### **Pregunta 3:**

¿Qué aportes y beneficios traerá la futura implantación de los módulos tutor y dominio de un sistema tutorial inteligente en la enseñanza de la IA 1 en la UCI?

## Anexo 2: Casos de uso del Sistema:

*Tabla 20:CU10: Listar tareas de aprendizaje*

|                    |   |
|--------------------|---|
| <b>CU- 10</b>      | <b>Listar tareas de aprendizaje</b>                                   |
| <b>Actor</b>       | Administrador   |
| <b>Descripción</b> | El administrador deberá listar todas las tareas que serán realizadas. |

*Tabla 21 CU11: Adicionar estudiante*

|                    |   |
|--------------------|---|
| <b>CU- 11</b>      | <b>Adicionar estudiante</b>   |
| <b>Actor</b>       | Profesor  |
| <b>Descripción</b> | El profesor deberá adicionar a los estudiantes que se incorporan en el sistema. |

*Tabla 22 CU-12: Gestionar Estudiante*

|                    |  |
|--------------------|--|
| <b>CU- 12</b>      | <b>Gestionar estudiante</b>  |
| <b>Actor</b>       | Administrador  |
| <b>Descripción</b> | El administrador realizará las tareas de eliminar, modificar, Buscar e insertar estudiantes. |