

**APLICACIÓN WEB PARA EL APRENDIZAJE DE ESTRUCTURAS
ARGUMENTATIVAS**

**LUIS GIOVANNY SUÁREZ THERÁN
OSCAR EDUARDO DÍAZ PARADA**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS
BUCARAMANGA**

2019

**APLICACIÓN WEB PARA EL APRENDIZAJE DE ESTRUCTURAS
ARGUMENTATIVAS**

**LUIS GIOVANNY SUÁREZ THERÁN
OSCAR EDUARDO DÍAZ PARADA**

**Trabajo de Grado para optar por el título de
Ingeniero de Sistemas e Informática**

**DIRECTORA
SONIA CRISTINA GAMBOA SARMIENTO
Doctora en Educación**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS
BUCARAMANGA**

2019

CONTENIDO

	Pág.
INTRODUCCIÓN	15
1. OBJETIVOS.....	17
1.1 OBJETIVO GENERAL	17
1.2 OBJETIVOS ESPECÍFICOS.....	17
2. MARCO DE REFERENCIA.....	18
2.1 FUNDAMENTOS TEÓRICOS.....	18
2.1.1 Argumentación.....	18
2.1.2 Modelo Argumentativo de Toulmin.	18
2.1.3 Entornos de Aprendizaje y TIC.	21
2.2 ENTORNOS DE DESARROLLO	22
2.2.1 Netbeans V 8.0.2.	22
2.2.2 Java Platform Enterprise Edition (JAVA EE) V 7.	23
2.2.3 MYSQL V 5.7.12.	23
2.3 FRAMEWORKS.....	24
2.3.1 Javasever Faces (JSF) V 2.2.....	24
2.3.2 Java Persistence API V 2.1.....	24
2.3.3 Enterprise Javabeans (EJB) V 3.1.	24
2.3.4 Bootstrap V 3.3.6.	25
2.3.5 Font Awesome V 4.5.0.....	25
3. METODOLOGÍA	26
3.1 ESPECIFICACIÓN DE REQUISITOS SOFTWARE	27
3.2 PROPÓSITO.....	27
3.3 ALCANCE DEL SISTEMA	27

3.4 DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS	28
3.5 PERSPECTIVA DEL PRODUCTO	28
3.6 FUNCIONES DEL PRODUCTO	28
3.7 RESTRICCIONES	29
3.8 SUPOSICIONES Y DEPENDENCIAS	29
3.9 REQUISITOS FUTUROS.....	29
4. REQUERIMIENTOS	30
4.1 REQUERIMIENTOS FUNCIONALES.....	30
4.2 REQUERIMIENTOS NO FUNCIONALES.....	32
4.3 ACTORES DEL SISTEMA	34
4.4 DESCRIPCIÓN DE ACTORES.....	34
4.5 PRINCIPALES CASOS DE USO	35
4.6 DIAGRAMA GENERAL DE CASOS DE USO Y ACTORES	36
4.7 DESCRIPCIÓN DE ACTORES	37
4.7.1 Administrador.....	37
4.7.2 Profesor	38
4.7.3 Estudiante	39
4.8 DESCRIPCIÓN DE DIAGRAMAS DE CASOS DE USO.....	39
4.8.1 Actualizar Teoría.....	39
4.8.2 Actualizar Ejemplo	41
4.8.3 Actualizar Ejercicio.....	42
4.8.4 Registrar Ejemplo	44
4.8.5 Registrar Ejercicio.....	45
4.8.6 Resolver Ejercicio	46
5. DEFINICIÓN DEL MODELO.....	47
5.1 DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS.....	48
5.2 DESARROLLO	50
5.3 PRUEBAS DE VERIFICACION	57

5.3.1 Primera Iteración.....	58
5.3.2 Segunda Iteración.....	58
5.3.3 Prototipo Final.....	59
5.4 DESCRIPCION DE PRUEBAS DE VERIFICACIÓN PARA EL PRODUCTO FINAL.....	60
5.4.1 Caso de Prueba: 01	60
5.4.2 Caso de Prueba: 02	61
5.4.3 Caso de Prueba: 03.....	61
5.4.4 Caso de Prueba: 04	62
5.4.5 Caso de Prueba: 05.....	62
5.4.6 Caso de Prueba: 06	62
5.4.7 Caso de Prueba: 07	63
5.4.8 Caso de Prueba: 08	64
5.4.8 Caso de Prueba: 09.....	64
6. CONCLUSIONES	65
BIBLIOGRAFÍA.....	66

LISTA DE TABLAS

	Pág.
Tabla 1. Requerimiento Funcional No. 1 – Interoperabilidad de usuarios del portal Filoen	30
Tabla 2. Requerimiento Funcional No. 2 – Gestionar teoría del Modelo de Toulmin	30
Tabla 3. Requerimiento Funcional No. 3 – Gestionar ejemplos del Modelo de Toulmin	31
Tabla 4. Requerimiento Funcional No. 4 – Gestionar ejercicios del Modelo de Toulmin	31
Tabla 5. Requerimiento Funcional No. 5 – Desarrollar ejercicios del Modelo de Toulmin	31
Tabla 6. Requerimiento Funcional No. 6 – Visualizar resultados.....	32
Tabla 7. Requerimiento No Funcional No. 1 – Interfaz del sistema	32
Tabla 8. Requerimiento No Funcional No. 2 – Diseño de la interfaz a semejanza del portal Filoen	32
Tabla 9. Requerimiento No Funcional No. 3 – Arquitectura de la aplicación web del Modelo de Toulmin	33
Tabla 10. Requerimiento No Funcional No. 4 – Ingreso de Datos.....	33
Tabla 11. Descripción de Actores – Administrador	34
Tabla 12. Descripción de Actores - Profesor	35
Tabla 13. Descripción de Actores – Estudiante	35
Tabla 14. Descripción de Caso de Uso – Actualizar Teoría.....	40
Tabla 15. Descripción de Caso de Uso – Actualizar Ejemplo	41
Tabla 16. Descripción de Caso de Uso – Actualizar Ejercicio	43
Tabla 17. Descripción de Caso de Uso – Registrar Ejemplo	44
Tabla 18. Descripción de Caso de Uso – Registrar Ejercicio.....	45
Tabla 19. Descripción de Caso de Uso – Resolver Ejercicio	46

Tabla 20. Resultados de las pruebas aplicadas la herramienta (iteración 1).....	58
Tabla 21. Resultados de las pruebas aplicadas la herramienta (iteración 2).....	58
Tabla 22. Resultados de las pruebas aplicadas a la interfaz de Portal (Producto Final).....	59
Tabla 23. Resultados de las pruebas aplicadas a la interfaz de Iniciar de Sesión (Producto Final).	59
Tabla 24. Resultados de las pruebas aplicadas a la interfaz de Registrarse (Producto Final).	59
Tabla 25. Resultados de las pruebas aplicadas la herramienta para el rol de Profesor (Producto Final).....	59
Tabla 26. Resultados de las pruebas aplicadas la herramienta para el rol de Estudiante (Producto Final).	60
Tabla 27. Caso de prueba 01	60
Tabla 28. Caso de prueba 02	61
Tabla 29. Caso de prueba 03	61
Tabla 30. Caso de prueba 04	62
Tabla 31. Caso de prueba 05	62
Tabla 32. Caso de prueba 06	63
Tabla 33. Caso de prueba 07	63
Tabla 34. Caso de prueba 08	64
Tabla 35. Caso de prueba 09	64

LISTA DE FIGURAS

	Pág.
Figura 1. Ejemplo de la estructura del modelo argumentativo de Toulmin.	20
Figura 2. Actores del sistema.....	34
Figura 3. Diagrama General de Casos de Uso y Actores	36
Figura 4. Diagrama de Caso de Uso - Administrador	37
Figura 5. Diagrama de Caso de Uso - Profesor	38
Figura 6. Diagrama de Caso de Uso - Estudiante.....	39
Figura 7. Diagrama de Caso de Uso – Actualizar Teoría.....	39
Figura 8. Diagrama de Caso de Uso – Actualizar Ejemplo	41
Figura 9. Diagrama de Caso de Uso – Actualizar Ejercicio	42
Figura 10. Diagrama de Caso de Uso – Registrar Ejemplo	44
Figura 11. Diagrama de Caso de Uso – Registrar Ejercicio.....	45
Figura 12. Diagrama de Caso de Uso – Resolver Ejercicio	46
Figura 13. Plantilla de recolección de información.....	47
Figura 14. Base de datos de la aplicación	48
Figura 15. Base de datos: Encargada del Modelo de Toulmin	49
Figura 16. Base de datos: Encargada del Inicio de la aplicación.....	49
Figura 17. Base de datos: Encargada de la información de los usuarios y sus relaciones	50
Figura 18. Primera iteración: Vista del Home	51
Figura 19. Primera iteración: Vista Agregar Ejercicios.....	51
Figura 20. Primera iteración: Vista Desarrollar Ejercicios	51
Figura 21. Segunda iteración: Vista Home	52
Figura 22. Segunda iteración: Vista de la Teoría	52
Figura 23. Segunda iteración: Vista de los ejercicios.....	53
Figura 24. Segunda iteración: Visualización Modificar Ejercicio	53
Figura 25. Tercera iteración: Vista actualizar teoría.....	54

Figura 26. Tercera iteración: Vista de los Ejemplos.....	54
Figura 27. Tercera iteración: Vista de un ejemplo.....	55
Figura 28. Tercera iteración: Vista modificar ejercicio	55
Figura 29. Tercera iteración: Visualización realizar Ejercicio	55
Figura 30. Visualización del backend, frontend y el persistence.....	57

RESUMEN

TÍTULO: APLICACIÓN WEB PARA EL APRENDIZAJE DE ESTRUCTURAS ARGUMENTATIVAS*

AUTORES: LUIS GIOVANNY SUÁREZ THERÁN, OSCAR EDUARDO DÍAZ PARADA**

PALABRAS CLAVE: FILOSOFÍA, MODELO DE TOULMIN, ESTRUCTURAS ARGUMENTATIVAS, APLICACIÓN WEB PARA LA ENSEÑANZA.

DESCRIPCIÓN:

En este informe se presenta el desarrollo de una aplicación web diseñada para facilitar el aprendizaje de estructuras argumentativas según el modelo propuesto por S. Toulmin. La aplicación web cuenta con una base de datos, desarrollada en MySQL, donde los profesores pueden almacenar argumentos, y su segmentación en elementos según el Modelo de Toulmin, de manera que los estudiantes pueden visualizarlos en forma de ejemplos y/o ejercicios que les permiten diferenciar los elementos del argumento según el Modelo Argumentativo propuesto por S. Toulmin. Así mismo, al resolver cada ejercicio, el estudiante podrá visualizar, de manera inmediata, la cantidad de elementos que respondió correctamente, y también le serán indicados los campos que fueron contestados de manera errónea, de manera adicional, el estudiante también podrá contrastar su respuesta con el ejercicio organizado de forma correcta, el cual se podrá apreciar justo debajo de la respuesta del estudiante, dado el caso que se presenten errores. La aplicación almacena las puntuaciones que obtienen los estudiantes al responder cada ejercicio, de manera que es posible, tanto para profesores como para el mismo estudiante, verificar su progreso, con el fin de examinar las posibles complicaciones que puedan presentar sus aprendices en la resolución de determinados ejercicios.

* Proyecto de grado.

** Facultad de Ingenierías Físico-Mecánicas, Escuela de Ingeniería de Sistemas e Informática. Directora: Sonia Cristina Gamboa Sarmiento, Doctora en Educación.

ABSTRACT

TITLE: WEB APPLICATION TO LEARN ARGUMENTATIVE STRUCTURES*

AUTHORS: LUIS GIOVANNY SUÁREZ THERÁN, OSCAR EDUARDO DÍAZ PARADA**

KEYWORDS: PHILOSOPHY, TOULMIN'S MODEL, ARGUMENTATIVE STRUCTURES, WEB APPLICATION FOR LEARNING.

DESCRIPTION:

In this document is shown the development of a web application designed to ease the learning of argumentative structures according to the model proposed by Stephen Toulmin. The web application has a data base, developed using MySQL, where it can store arguments, and its segmentation in elements according to the Toulmin Model of argument, in a way that students can visualize them as examples and/or exercises allowing them to differentiate between the elements of an argument as stated in the argumentative model proposed by S. Toulmin. Likewise, when solving each exercise, the student will be able to visualize, at once, the amount of elements that were answered correctly, and also the fields that were answered in an erroneous way, will be indicated, in addition, the student will also be able to contrast his answer with the exercise organized correctly, which can be seen just below the response given by the student, if there are errors. The web application stores the scores from the students' answers on each exercise, so that it is possible, both for teachers and students, to verify their progress, in order to examine the possible complications that their apprentices may present in the resolution of certain exercises.

* Bachelor Thesis.

** Facultad de Ingenierías Físico-Mecánicas, Escuela de Ingeniería de Sistemas e Informática. Directora: Sonia Cristina Gamboa Sarmiento, Doctora en Educación.

INTRODUCCIÓN

Filosofía y enseñanza de la filosofía es un grupo de investigación interinstitucional, conformado por investigadores que pertenecen a universidades públicas de Colombia y Estados Unidos que pone en el centro de sus estudios: el mundo de la vida, la constitución de la identidad personal y la identidad latinoamericana. Desarrollan investigación básica en fenomenología, hermenéutica y sobre el concepto de formación; y desarrollan investigación aplicada a la enseñanza de la filosofía, con especial interés de la misma en entornos virtuales.

Para la práctica de la filosofía, la argumentación hace parte de las habilidades propias del filosofar. El desarrollo de dicha habilidad implica tanto el conocimiento de las teorías de la argumentación como estructura lógica formal, así como la práctica en la construcción y reconocimiento de argumentos para determinar su validez y su plausibilidad. Sobre la argumentación como estructura los aportes teóricos más reconocidos en el ámbito son la *Retórica*, de Aristóteles (1990), *El imperio retórico*, de Ch. Perelman (1977), el *Tratado de la argumentación*, de Perelman y Olbrechts-Tyteca (1958), y *Los usos de la argumentación* de Stephen Toulmin (2007), entre otros. Específicamente, el Modelo propuesto por Stephen Toulmin permite reconocer en un discurso una estructura que permite determinar si se trata de un argumento, cuál es su tesis o conclusión, y cuáles las premisas que lo sustentan, de manera que, en consecuencia, se pueda establecer si tal conclusión está argumentada, para establecer un nivel de plausibilidad.

Sobre estos referentes teóricos, el grupo Filosofía y enseñanza de la filosofía ha venido desarrollando herramientas software para el aprendizaje de la lógica formal, de silogismos y de argumentación, como habilidades propias del filosofar. Dentro del desarrollo de diferentes herramientas y aplicaciones software bajo sus énfasis,

encontramos la herramienta “Automatización del Modelo de Toulmin”¹ que le permite al usuario establecer los componentes de un argumento, según el Modelo de S. Toulmin, y determinar si lo hizo adecuadamente; esta funcionalidad se lleva a cabo con base en registros que almacenan argumentos ya construidos y caracterizados. Por esta razón, se desarrolló una nueva versión de esta herramienta que plantea una estructura sintáctica de representación de argumentos cuya automatización permite clasificar las partes de un argumento cualquiera, en los componentes propuestos por S. Toulmin.

¹ TOLEDO PARRA, Camilo Andrés. Automatización del Modelo de Toulmin. Bucaramanga. 2012. Tesis Pregrado. Universidad Industrial de Santander.

1. OBJETIVOS

1.1 OBJETIVO GENERAL

Diseñar e implementar una aplicación web para el apoyo del aprendizaje de las estructuras argumentativas propuestas por el Modelo de Toulmin, mediante la exposición de conceptos teóricos y la realización de actividades prácticas de tal modelo.

1.2 OBJETIVOS ESPECÍFICOS

- Representar el Modelo de Toulmin como una estructura de datos que permita almacenar, modificar y consultar argumentos.
- Diseñar y desarrollar los módulos de la aplicación que permitan:
 - Administración de contenido (teoría, ejemplos y ejercicios). Actualización y registro de teoría y ejemplos, además de ejercicios de diferente nivel de dificultad propuestos por profesores.
 - Interoperabilidad de usuarios con el portal Filoen.
 - Administración de permisos de usuario que definen el acceso a las diferentes funcionalidades de la aplicación.
 - Generación de reportes.
- Implementar la aplicación en el portal web para la enseñanza de la filosofía “Filoen²”.

² Portal para la enseñanza de la filosofía diseñado y desarrollado por el Grupo Filosofía y enseñanza de la filosofía [FiloEn] <<http://filosofiyensenanza.uis.edu.co:8080/portalfiloen>>.

2. MARCO DE REFERENCIA

2.1 FUNDAMENTOS TEÓRICOS

La *argumentación* es una habilidad propia del filosofar, por lo que los estudiantes de filosofía deben contar con espacios curriculares que les permitan el aprendizaje, tanto teórico como práctico de esta habilidad. Se sabe, por otra parte, que la *didáctica* implica que el aprendizaje se ve favorecido si se ejerce en entornos que resultan naturales para los estudiantes, y para ello las TIC, en especial los entornos web, representan no solo un entorno natural, sino adecuado para lograr una interacción entre estudiantes y profesores. A continuación, se exponen los principales referentes teóricos de este proyecto.

2.1.1 Argumentación. La definición formal de argumento según la Real Academia Española (RAE) “es el razonamiento que se emplea para probar o demostrar una proposición, o bien para convencer a alguien de aquello que se afirma o se niega”³.

Ahora bien, la *argumentación* según Toulmin, “consiste en el conjunto de los argumentos justificatorios utilizados para apoyar afirmaciones, en las estructuras que pueden tener, en el valor que puede reivindicar para sí y en el modo en que nos enfrentamos a ellos al clasificarlos”⁴.

2.1.2 Modelo Argumentativo de Toulmin. El *modelo de Toulmin* propone una representación de elementos invariantes en el proceso de construcción de un argumento, cuya estructura está determinada por la dependencia que se establece

³ REAL ACADEMIA ESPAÑOLA. Diccionario de la lengua española. En: Argumento [en línea]. 23 Ed. (2014). <<http://dle.rae.es/?id=3YKtkpX&o=h>>.

⁴ GAMBOA SARMIENTO, Sonia. Formación y argumentación: automatización de procesos argumentativos. Bucaramanga, 2011. Tesis doctoral (Doctorado Interinstitucional en Educación). Universidad Pedagógica Nacional. p. 56.

entre el valor de una afirmación y los “fundamentos o razones en los que se apoya, datos, hechos, pruebas, consideraciones, componentes”⁵.

Toulmin propone que un argumento sustantivo va desde los datos (D) hasta la conclusión (C), donde D corresponde a la información, antecedentes o hechos de los cuales disponemos para dar fundamento a C. El modelo también incorpora garantías (W), apoyo (B), calificador modal (Q) y las condiciones de refutación (R). Además, Toulmin supone que un argumento propiamente dicho consiste en, al menos, tres componentes esenciales: D, C y W⁶.

Así, tenemos que la estructura de este modelo consta de seis categorías⁷:

- **Garantía (Warrant):** Es una licencia para inferir, una ley de pasaje, que permite el paso de los datos a las conclusiones.
- **Apoyo, respaldo o soporte (Backing):** Es el cuerpo de información general que apoya a la garantía y al dato, y que nos remite al mundo sustancial en el que nos encontramos.
- **Datos (Datum):** Son hechos o evidencias del mundo empírico que se entrega a favor de la conclusión.
- **Conclusiones (Claims):** Es aquello que se sostiene, el enunciado que se justifica a partir de la garantía y del dato. En otras palabras, corresponden a las pretensiones, demandas o alegatos que buscan, entre otras cosas, posicionar una acción, una perspectiva. En síntesis, dado el dato y aceptada la garantía, ambos apoyados en sus respectivos respaldos, se obtiene la conclusión.
- **Cualificadores o modalizadores (Qualifiers o modality):** Son aquellas construcciones lingüísticas que permiten atenuar una pretensión, por ejemplo: siempre, a veces, probablemente, depende, etc.

⁵ Ibid., p. 55.

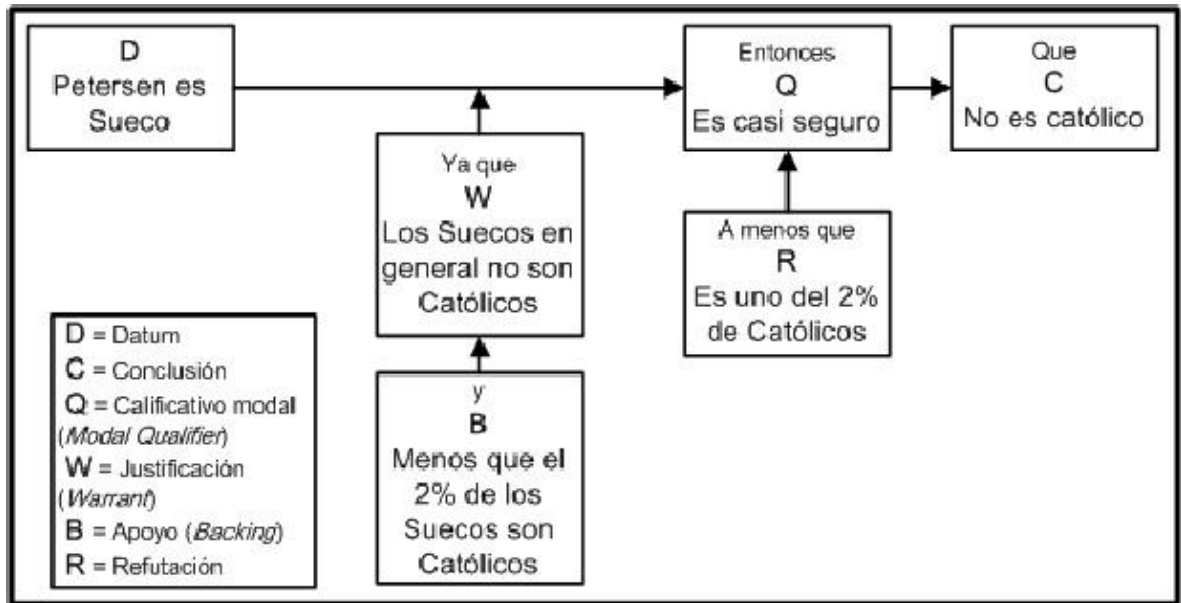
⁶ PINOCHET, Jorge. El modelo argumentativo de Toulmin y la educación en ciencias: una revisión argumentada. En: *Ciência & Educação* (Bauru). Vol. 21, no. 2, p. 311.

⁷ MEZA, Paulina. Aproximación al modelo argumentativo de Stephen Toulmin mediante su aplicación a cartas de opinión. En: *Simpósio Internacional de Estudos de Gêneros Textuais* (V : 2009 : Caxias do Sul). p 4-5

- Refutaciones o restricciones condiciones de refutación (Rebuttals): Corresponden a aquellas circunstancias excepcionales que pueden socavar la fuerza de los argumentos. Son, en otras palabras, las condiciones de refutación que la conclusión permite.

Para el siguiente ejemplo “Petersen es sueco, y dado que los suecos en general no son católicos, pues menos del 2% de los suecos son católicos, es casi seguro que Petersen no es católico, a menos que sea uno del 2% que es católico.”. Este ejemplo puede representarse gráficamente del siguiente modo:

Figura 1. Ejemplo de la estructura del modelo argumentativo de Toulmin.



Fuente: TRYSTAN-ZICKLER, Ashlar. Acerca del modelo de Toulmin de argumentación: aplicaciones y debates.

Así, las garantías (warrants) son las que permiten justificar y legitimar el paso desde los datos (datum) a la conclusión (claims). El modelo incorpora explícitamente el grado de certeza (o incerteza) del argumento mediante el calificador modal Q (qualifier). Además, el modelo muestra las situaciones bajo las cuales C no sería válida, también conocidas como las condiciones de refutación (rebuttals), que

establecen las restricciones que se aplican a C. Por último, el apoyo (backings) se refiere a las circunstancias generales bajo las cuales W es apoyada⁸.

El Modelo de Toulmin ofrece una comprensión adicional de la argumentación silogística, en la que considera que los argumentos pueden estar influidos por matizantes que permiten que aun cuando la conclusión no sea contundente, puede ser creíble en cuanto su estructura este bien fundamentada por sus componentes. Por otra parte, Toulmin ofrece una sistematización de estas estructuras argumentativas que facilitan su comprensión y evaluación, precisamente, para adherirse a la conclusión o rechazarla⁹. Este Modelo, por tanto, representa un recurso didáctico excepcional para la enseñanza de la argumentación. Lograr, como pretende este proyecto que, además, su evaluación se haga automáticamente, representa para el estudiante la posibilidad de estudiar estas estructuras con autonomía y al ritmo de su voluntad.

2.1.3 Entornos de Aprendizaje y TIC. Los entornos de aprendizaje se definen como un conjunto de experiencias que permiten que el estudiante (para su formación personal, investigativa o docente), de manera planeada y estructurada, ejerza procesos autónomos de construcción de conocimiento propios de una disciplina y, preferencialmente, habilidades de conocimiento dentro de ellas.

Los entornos de aprendizaje están conformados, entre otros elementos, por: las personas (profesores, estudiantes, expertos), los materiales, las bibliotecas, la interacción personal (entre estudiantes, profesores-estudiantes, etc.), el texto, la información en línea, entornos virtuales, carteleras.

⁸ PINOCHET, Jorge. El modelo argumentativo de Toulmin y la educación en ciencias: una revisión argumentada. En: *Ciência & Educação* (Bauru). Vol. 21, no. 2, p. 311.

⁹ GAMBOA SARMIENTO, Sonia. Formación y argumentación: automatización de procesos argumentativos. Bucaramanga, 2011. Tesis doctoral (Doctorado Interinstitucional en Educación). Universidad Pedagógica Nacional. p. 87.

Los entornos virtuales de aprendizaje, particularmente, constituyen una alternativa adecuada para satisfacer los requerimientos de los estudiantes, dado que estos ofrecen ventajas que potencian el aprendizaje con sentido analítico y crítico. Tales como¹⁰:

- Favorecen el aprendizaje colaborativo, ya que permiten la discusión de textos entre estudiantes y entre estos con los profesores.
- Permite implementar estrategias didácticas que fomenten el desarrollo de las capacidades de análisis, de crítica y de interpretación de los estudiantes (como la lectura comprensiva de textos, redacción de ensayos, el comentario y la reseña, entre otros).
- Permiten mayor acceso del estudiante a la ejercitación y el análisis individual del material de estudio, independiente del tiempo y espacio destinado para la clase formal.

2.2 ENTORNOS DE DESARROLLO

2.2.1 Netbeans V 8.0.2. NetBeans es un entorno de desarrollo integrado libre (IDE)¹¹, hecho principalmente para el lenguaje de programación Java con soporte en este. *NetBeans IDE* es un producto multiplataforma, libre y gratuito sin restricciones de uso.

La plataforma *NetBeans* permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos, los cuales contienen clases de java escritas para interactuar con las APIs¹² de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la

¹⁰ VARGAS, Germán; GAMBOA, Sonia y REEDER, Harry. La humanización como formación: La filosofía y la enseñanza de la filosofía en la condición postmoderna. 1 ed. Bogotá. San Pablo, 2008, p. 156-157.

¹¹ Integrated Development Environment (IDE), es una aplicación informática que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de software.

¹² La interfaz de programación de aplicaciones (API) es el conjunto de subrutinas, funciones y procedimientos (métodos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software¹³.

2.2.2 Java Platform Enterprise Edition (JAVA EE) V 7. *Java Platform, Enterprise Edition (EE)* es una plataforma para desarrollar y ejecutar software de aplicaciones en el lenguaje de programación Java.

Java EE tiene varias especificaciones de API, tales como JDBC, RMI, e-mail, JMS, Servicios Web, etc y define cómo coordinarlos. Java EE permite al desarrollador crear una aplicación de empresa portable entre plataformas y escalable, a la vez que integrable con tecnologías anteriores. Otros beneficios añadidos son, por ejemplo, que el servidor de aplicaciones puede manejar transacciones, la seguridad, escalabilidad, concurrencia y gestión de los componentes desplegados, significando que los desarrolladores pueden concentrarse más en la lógica de negocio de los componentes en lugar de en tareas de mantenimiento de bajo nivel. También es posible empezar con poco o ningún coste¹⁴.

2.2.3 MYSQL V 5.7.12. MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario. Además, es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones¹⁵.

¹³ WIKIPEDIA. Netbeans [en línea]. <<https://es.wikipedia.org/wiki/NetBeans>>.

¹⁴ WIKIPEDIA. Java EE [en línea]. <https://es.wikipedia.org/wiki/Java_EE>.

¹⁵ WIKIPEDIA. MySQL [en línea]. <<https://es.wikipedia.org/wiki/MySQL>>.

2.3 FRAMEWORKS

2.3.1 Javaser Faces (JSF) V 2.2. JavaServer Faces (*JSF*) es una tecnología y framework para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE. JSF usa JavaServer Pages (JSP) como la tecnología que permite hacer el despliegue de las páginas, pero también se puede acomodar a otras tecnologías como XUL (acrónimo de XML-based User-interface Language, lenguaje basado en XML para la interfaz de usuario)¹⁶.

JSF está incluida en la plataforma Java Enterprise Edition (Java EE), que permite crear aplicaciones que usen JSF sin necesidad de agregar alguna librería extra en el proyecto.

2.3.2 Java Persistence API V 2.1. Java Persistence API, más conocida por sus siglas *JPA*, es la API de persistencia desarrollada para la plataforma Java EE. Es un framework del lenguaje de programación Java que maneja datos relacionales en aplicaciones usando la Plataforma Java en sus ediciones Standard (Java SE) y Enterprise (Java EE)¹⁷.

El objetivo que persigue el diseño de esta API es no perder las ventajas de la orientación a objetos al interactuar con una base de datos (siguiendo el patrón de mapeo objeto-relacional).

2.3.3 Enterprise Javabeans (EJB) V 3.1. Los *EJB* proporcionan un modelo de componentes distribuido estándar del lado del servidor. El objetivo de los EJB es dotar al programador de un modelo que le permita abstraerse de los problemas generales de una aplicación empresarial (conurrencia, transacciones, persistencia, seguridad, etc.) para centrarse en el desarrollo de la lógica de negocio en sí. El

¹⁶ WIKIPEDIA. JavaServer Faces [en línea]. <https://es.wikipedia.org/wiki/JavaServer_Faces>.

¹⁷ WIKIPEDIA. Java Persistence API [en línea]. <https://es.wikipedia.org/wiki/Java_Persistence_API>.

hecho de estar basado en componentes permite que éstos sean flexibles y sobre todo reutilizables¹⁸.

Su especificación detalla cómo los servidores de aplicaciones proveen objetos desde el lado del servidor, que son precisamente los EJB:

- Comunicación remota utilizando CORBA.
- Transacciones.
- Control de la concurrencia.
- Servicios de nombres y de directorio.
- Seguridad.
- Ubicación de componentes en un servidor de aplicaciones.

2.3.4 Bootstrap V 3.3.6. Bootstrap es un framework CSS que permite dar forma a un sitio web mediante librerías CSS que incluyen tipografías, botones, cuadros, menús y otros elementos que pueden ser utilizados en cualquier sitio web.

Bootstrap es una excelente herramienta para crear interfaces de usuario limpias y totalmente adaptables a todo tipo de dispositivos y pantallas, sea cual sea su tamaño. Además, Bootstrap ofrece las herramientas necesarias para crear cualquier tipo de sitio web utilizando los estilos y elementos de sus librerías¹⁹.

2.3.5 Font Awesome V 4.5.0. *Font Awesome* es un framework que se usa para implementar iconos sin necesidad de imágenes y sin técnicas intrusivas.

La mayoría de iconos que se pueden usar con Font Awesome están relacionados a acciones que regularmente se llevan a cabo en sitios o aplicaciones web²⁰.

¹⁸ WIKIPEDIA. Enterprise JavaBeans [en línea]. <https://es.wikipedia.org/wiki/Enterprise_JavaBeans>.

¹⁹ FONTELA, Alvaro. ¿Qué es Bootstrap? [en línea]. <<https://raiolanetworks.es/blog/que-es-bootstrap/>>

²⁰ Iel, Dan. Usando íconos sin imágenes con Font Awesome [en línea]. <<http://maquetando.com/recursos/fonts/usando-icnos-sin-imagenes-con-font-awesome/>>.

3. METODOLOGÍA

La metodología escogida para el desarrollo de la aplicación fue el *modelo de prototipos*, dado que nuestro cliente (Grupo Filosofía y enseñanza de la filosofía) conocía los objetivos generales para el software, pero no identificaba los requisitos detallados de entrada, procesamiento o salida. Además, permitió que la aplicación se construyera gracias a la facilidad de comprensión y aclaración de ciertos aspectos en los que el cliente tuviera claro lo que necesitaba así como también la solución que se propuso para dicha necesidad y de esta forma minimizar el riesgo y la incertidumbre en el desarrollo mediante la visibilidad del producto desde el inicio del ciclo de vida con el primer prototipo, ayudando al cliente a definir mejor los requisitos y a ver las necesidades reales del producto.

Siguiendo esta metodología, se realizaron tres iteraciones principalmente para el desarrollo de esta aplicación, y son las siguientes:

- Para la primera iteración se aclaró la estructura del Modelo de Toulmin como única por ambas partes y de esta manera se procedió a un desarrollo rápido de esta. El cual consistió en unas plantillas donde los profesores ingresarán uno o más argumentos, constituidos por la estructura definida previamente del Modelo de Toulmin.
- Durante la segunda iteración, partiendo de la plantilla antes definida, se diseñó e implantó una base de datos del modelo para ejemplos y ejercicios, además de la información principal del Modelo, que permite la administración del contenido del aplicativo.
- Para la tercera y última iteración, se corrobora la interoperabilidad de usuarios con el portal Filoen, donde pondrá en marcha la BBDD creada, donde los usuarios (tanto estudiantes como profesores) podrán acceder a la información del Modelo de Toulmin y utilizar esta herramienta como práctica y enseñanza para su aprendizaje.

3.1 ESPECIFICACIÓN DE REQUISITOS SOFTWARE

La Especificación de Requisitos Software (ERS) de la aplicación web para el apoyo del aprendizaje de las estructuras argumentativas propuestas por el Modelo de Toulmin se llevó a cabo con la participación de profesores del Grupo de filosofía y enseñanza de la filosofía, del cual hace parte el presente proyecto.

Este documento se ha realizado en base a las directrices dadas por el estándar “IEEE Recommended Practice for Software Requirement Specifications ANSI/IEEE 830 1998”.

3.2 PROPÓSITO

El propósito general de la aplicación web es el apoyo del aprendizaje de las estructuras argumentativas propuestas por el Modelo de Toulmin. Por consiguiente, se desarrolló la aplicación, que cumple con todos los requisitos que se establecieron en el ERS, en el cual se definió de manera clara y precisa todas las funcionalidades y restricciones del sistema que se deseó construir.

3.3 ALCANCE DEL SISTEMA

En la actualidad, el grupo de Filosofía y enseñanza de la filosofía, cuenta con una herramienta pedagógica para la enseñanza de figuras retóricas, sin embargo, no cuenta con una herramienta interactiva que permita el desarrollo cognitivo de la habilidad propia de argumentar, como lo es el modelo de S. Toulmin. La aplicación web para el aprendizaje de estructuras argumentativas, tiene como objetivo principal facilitar la realización de las siguientes funciones:

- Gestión de teoría propia del modelo de S. Toulmin.
- Gestión de ejemplos propuesto por docentes expertos en el área.
- Gestión de ejercicios basados en la estructura propia del modelo de S. Toulmin.

3.4 DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS

HTTP: Hypertext Transfer Protocol.

JSF: Java Server Faces.

RF: Requisitos Funcionales.

RNF: Requisitos No Funcionales.

Casos de Uso: Descripción de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso.

3.5 PERSPECTIVA DEL PRODUCTO

La aplicación web para el apoyo del aprendizaje del modelo de Toulmin será un producto diseñado para trabajar en entornos WEB, lo que generará una utilización de forma rápida y eficaz. Su desarrollo está soportado en un servidor WEB, WildFly, y una base de datos relacional, MYSQL Workbench, que permitirán el acceso a los servicios del sistema.

3.6 FUNCIONES DEL PRODUCTO

Gestión de Teoría: El sistema permite registrar y actualizar la teoría referente al Modelo de S. Toulmin.

Gestión de ejemplos: El sistema permite el registro, actualización y eliminación de ejemplos referentes al modelo argumentativo propuesto por S. Toulmin.

Gestión de ejercicios: El sistema permite el registro, actualización y eliminación de ejercicios estructurados según el modelo argumentativo de S. Toulmin.

Interoperabilidad de usuarios con el portal filoen: El sistema permite el uso compartido de registro y gestión de usuarios presentes en el portal filoen.

3.7 RESTRICCIONES

- La disponibilidad del sistema debe ser absoluta, excepto en el caso de cortes en la red de comunicaciones.
- Lenguaje y tecnologías en uso JAVA (JSF).
- Se desarrollará en NetBeans IDE.
- Se usa una base de datos relacional: MySql
- Lenguaje de marcado XHTML.
- Framework web Prime Faces.

3.8 SUPOSICIONES Y DEPENDENCIAS

- Se asume que habrá una conexión a internet en los equipos donde se desee acceder a la aplicación.
- La información que se presenta inicialmente la aplicación web sobre el modelo de Toulmin fue suministrada por diferentes docentes conocedores del tema.

3.9 REQUISITOS FUTUROS

- Creación de grupos de profesor y estudiantes.
- Versión móvil para Smartphone y Tablet.

4. REQUERIMIENTOS

4.1 REQUERIMIENTOS FUNCIONALES

Tabla 1. Requerimiento Funcional No. 1 – Interoperabilidad de usuarios del portal Filoen

Identificación del requerimiento:	RF01
Nombre del Requerimiento:	Interoperabilidad de usuarios del portal Filoen.
Características:	El sistema presentará la interoperabilidad de usuarios (autenticación y registro de usuarios) con el portal Filoen.
Descripción del requerimiento:	El sistema debe permitir la conexión al portal “Filoen”, con el cual permitirá la interoperabilidad de los usuarios. Dependiendo del rol del usuario se define el acceso a las diferentes funcionalidades de la aplicación.
Requerimiento NO funcional:	<ul style="list-style-type: none">• RNF01• RNF02
Prioridad del requerimiento:	Alta

Tabla 2. Requerimiento Funcional No. 2 – Gestionar teoría del Modelo de Toulmin

Identificación del requerimiento:	RF02
Nombre del Requerimiento:	Gestionar teoría del Modelo de Toulmin.
Características:	El sistema permitirá registrar y actualizar la teoría del Modelo de Toulmin.
Descripción del requerimiento:	Permite a los roles administrador y profesor registrar y actualizar el contenido de Teoría.
Requerimiento NO funcional:	<ul style="list-style-type: none">• RNF01• RNF02
Prioridad del requerimiento:	Alta

Tabla 3. Requerimiento Funcional No. 3 – Gestionar ejemplos del Modelo de Toulmin

Identificación del requerimiento:	RF03
Nombre del Requerimiento:	Gestionar ejemplos del Modelo de Toulmin.
Características:	El sistema permitirá registrar, actualizar y eliminar ejemplos del Modelo de Toulmin.
Descripción del requerimiento:	Permite a los roles administrador y profesor registrar, actualizar y eliminar Ejemplos del Modelo de Toulmin.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF02 • RNF04
Prioridad del requerimiento: Alta	

Tabla 4. Requerimiento Funcional No. 4 – Gestionar ejercicios del Modelo de Toulmin

Identificación del requerimiento:	RF04
Nombre del Requerimiento:	Gestionar ejercicios del Modelo de Toulmin.
Características:	El sistema permitirá registrar, actualizar y eliminar ejercicios del Modelo de Toulmin.
Descripción del requerimiento:	Permite a los roles administrador y profesor registrar, actualizar y eliminar ejercicios del Modelo de Toulmin.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF02 • RNF04
Prioridad del requerimiento: Alta	

Tabla 5. Requerimiento Funcional No. 5 – Desarrollar ejercicios del Modelo de Toulmin

Identificación del requerimiento:	RF05
Nombre del Requerimiento:	Desarrollar ejercicios del Modelo de Toulmin.
Características:	El sistema permitirá desarrollar ejercicios del Modelo de Toulmin.
Descripción del requerimiento:	Permite al rol estudiante realizar ejercicios del Modelo de Toulmin propuestos por los profesores y ver su respectivo resultado.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF02
Prioridad del requerimiento: Alta	

Tabla 6. Requerimiento Funcional No. 6 – Visualizar resultados

Identificación del requerimiento:	RF06
Nombre del Requerimiento:	Visualizar resultados según el puntaje obtenido de los ejercicios del Modelo de Toulmin.
Características:	El sistema permite visualizar la cantidad de ejercicios resueltos así como el número de veces en las que se han obtenido resultados superiores a 4/6 e inferiores a 2/6 del Modelo de Toulmin.
Descripción del requerimiento:	Permite a los roles administrador y profesor visualizar la cantidad de veces que se han resuelto los ejercicios, además de la cantidad de veces que las puntuaciones han sido superiores a 4/6 e inferiores a 2/6 de los ejercicios del Modelo de Toulmin.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF02
Prioridad del requerimiento: Alta	

4.2 REQUERIMIENTOS NO FUNCIONALES

Tabla 7. Requerimiento No Funcional No. 1 – Interfaz del sistema

Identificación del requerimiento:	del	RNF01
Nombre del Requerimiento:	del	Interfaz del sistema.
Características:		El sistema presentará una interfaz de usuario simple para que sea de fácil manejo a los usuarios del sistema.
Descripción del requerimiento:	del	El sistema debe tener una interfaz de uso intuitiva y sencilla.
Prioridad del requerimiento: Alta		

Tabla 8. Requerimiento No Funcional No. 2 – Diseño de la interfaz a semejanza del portal Filoen

Identificación del requerimiento:	RNF02
Nombre del Requerimiento:	Diseño de la interfaz a semejanza del portal Filoen.
Características:	El sistema deberá de tener una interfaz de usuario, teniendo en cuenta las características del portal Filoen.
Descripción del requerimiento:	La interfaz de usuario debe ajustarse a las características del portal Filoen, dentro de la cual estará incorporado la aplicación web para el aprendizaje de estructuras argumentativas.
Prioridad del requerimiento: Alta	

Tabla 9. Requerimiento No Funcional No. 3 – Arquitectura de la aplicación web del Modelo de Toulmin

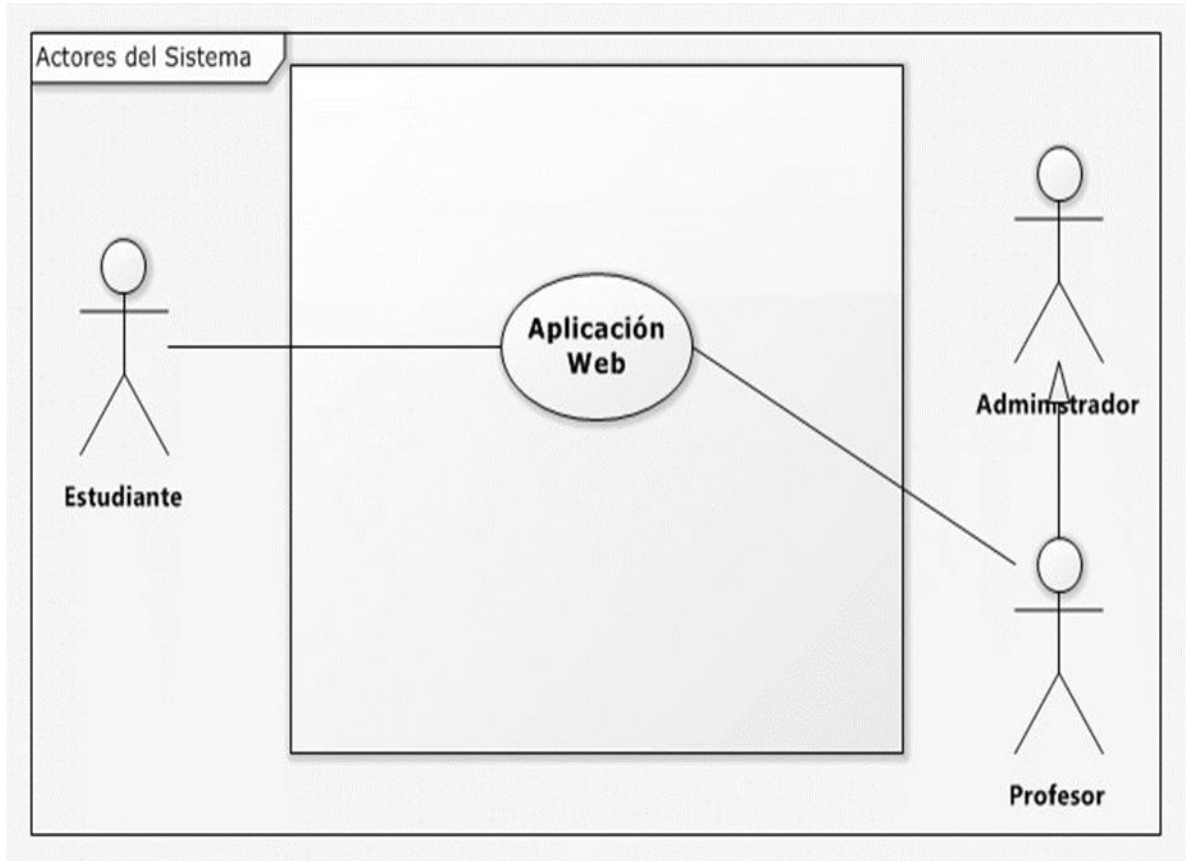
Identificación del requerimiento:	RNF03
Nombre del Requerimiento:	Arquitectura de la aplicación web del Modelo de Toulmin.
Características:	El sistema será desarrollado en la web y toda la administración se realizará desde un navegador.
Descripción del requerimiento:	El sistema tendrá interfaces gráficas de administración y de operación en ambiente Web, para permitir su utilización a través de exploradores o navegadores de Internet.
Prioridad del requerimiento: Alta	

Tabla 10. Requerimiento No Funcional No. 4 – Ingreso de Datos

Identificación del requerimiento:	RNF04
Nombre del Requerimiento:	Ingreso de Datos.
Características:	Los datos que se ingresen a la estructura de datos deben ser de manera controlada para evitar que los recursos del servidor de la base de datos se pongan al máximo nivel.
Descripción del requerimiento:	El ingreso de datos al sistema se hará por medio de transacciones que permitan el ingreso de los datos de forma parcial; es decir, que el tamaño de las páginas de los formularios sea el adecuado.
Prioridad del requerimiento: Alta	

4.3 ACTORES DEL SISTEMA

Figura 2. Actores del sistema



4.4 DESCRIPCIÓN DE ACTORES

Tabla 11. Descripción de Actores – Administrador

Actor	Administrador
Caso de Uso	Actualizar Teoría, Actualizar Ejemplos, Actualizar Ejercicios, Registrar Ejemplos y Registrar Ejercicios.
Descripción	Un actor denominado como Administrador tiene el alcance de gestionar todo, es decir, registrar Ejemplos y Ejercicios así como actualizarlos, al igual que actualizar la Teoría. Además, puede ver el historial de resultados obtenidos por los estudiantes tras resolver ejercicios.
Tipo	Principal

Tabla 12. Descripción de Actores - Profesor

Actor	Profesor
Caso de Uso	Actualizar Teoría, Actualizar Ejemplos, Actualizar Ejercicios, Registrar Ejemplos y Registrar Ejercicios.
Descripción	Un actor denominado como Profesor tiene el permiso de registrar Ejemplos y Ejercicios así como actualizarlos, al igual que actualizar la Teoría. Adicionalmente, puede ver el historial de los resultados obtenidos por los estudiantes al resolver ejercicios.
Tipo	Principal

Tabla 13. Descripción de Actores – Estudiante

Actor	Estudiante
Caso de Uso	Resolver Ejercicio.
Descripción	Un usuario estudiante puede resolver ejercicios y ver el respectivo resultado.
Tipo	Principal

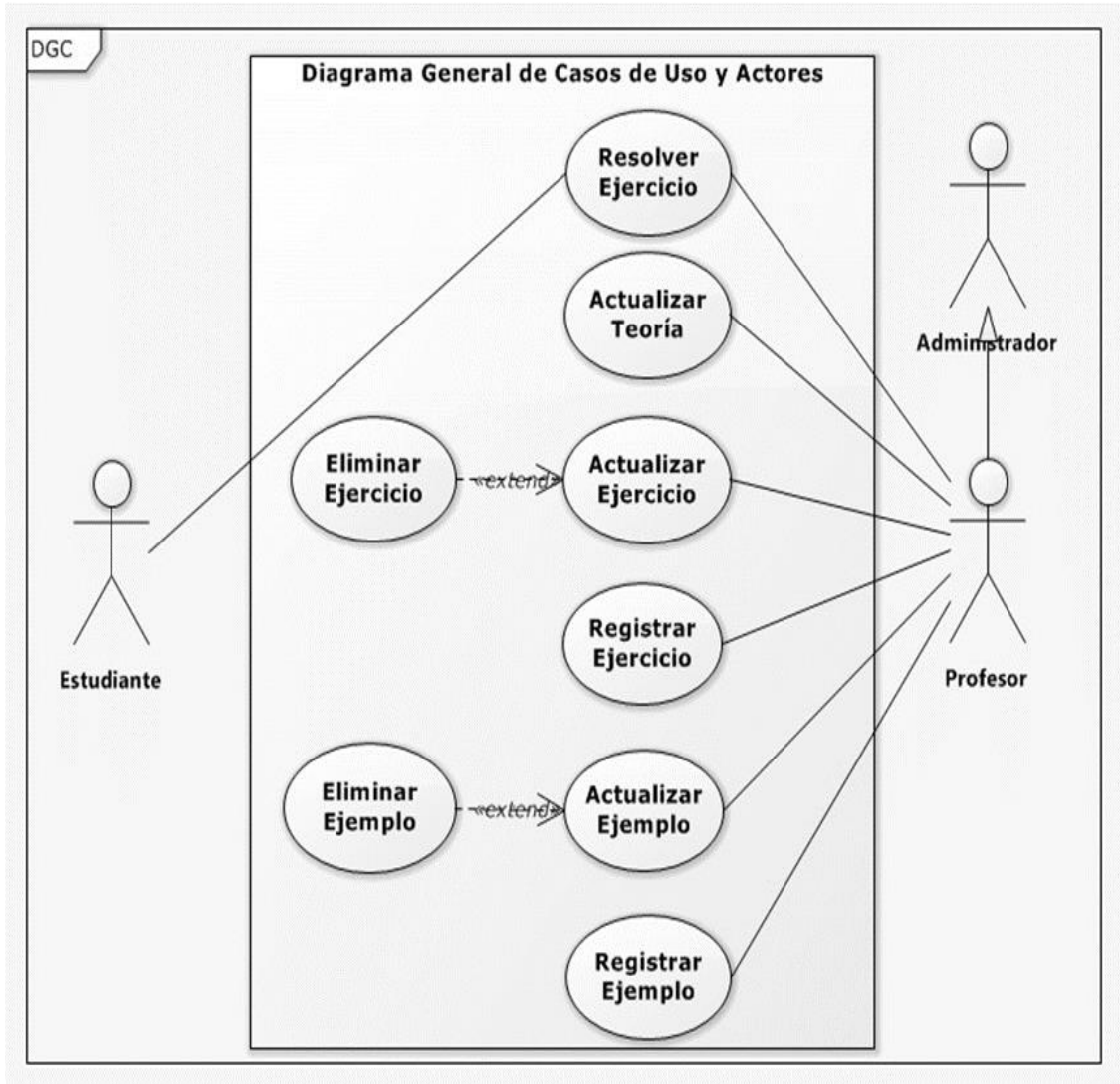
4.5 PRINCIPALES CASOS DE USO

Una vez identificados los actores del sistema, se procede a identificar y diseñar los principales casos de uso del sistema a desarrollar. Los casos de uso identificados son:

- 1) Actualizar Teoría.
- 2) Actualizar Ejemplo.
- 3) Actualizar Ejercicio.
- 4) Registrar Ejemplo.
- 5) Registrar Ejercicio.
- 6) Resolver Ejercicio.

4.6 DIAGRAMA GENERAL DE CASOS DE USO Y ACTORES

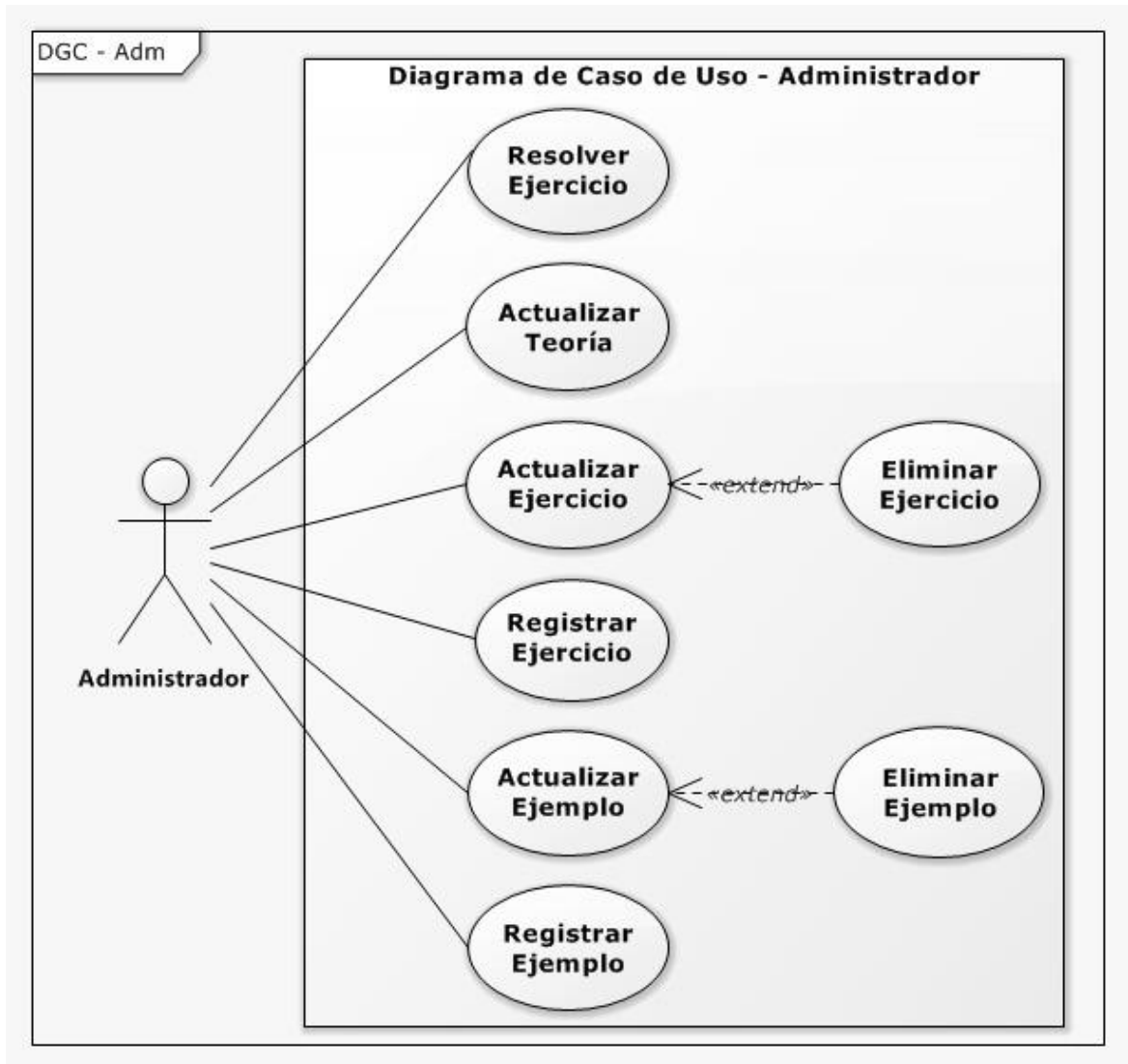
Figura 3. Diagrama General de Casos de Uso y Actores



4.7 DESCRIPCIÓN DE ACTORES

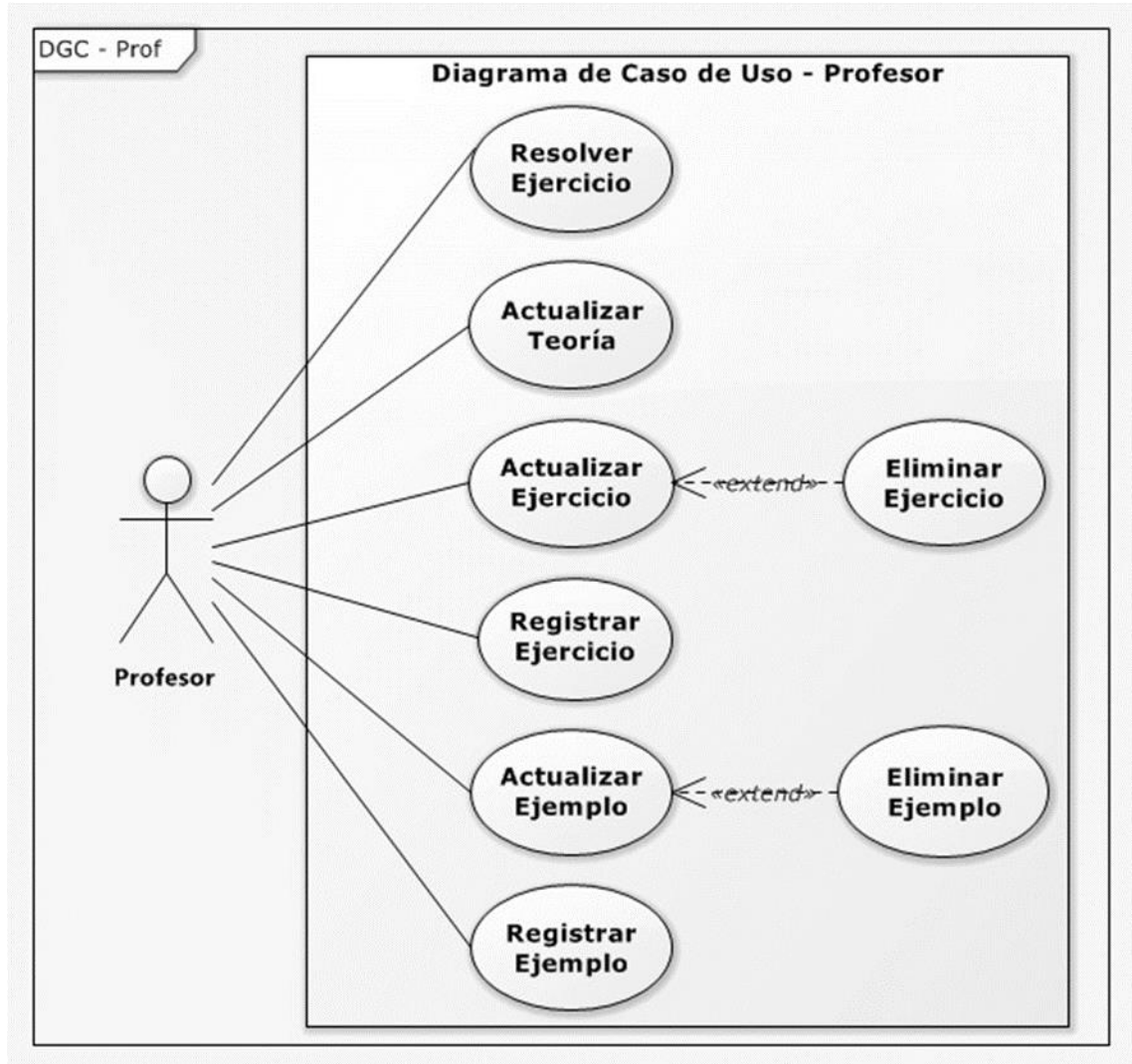
4.7.1 Administrador

Figura 4. Diagrama de Caso de Uso - Administrador



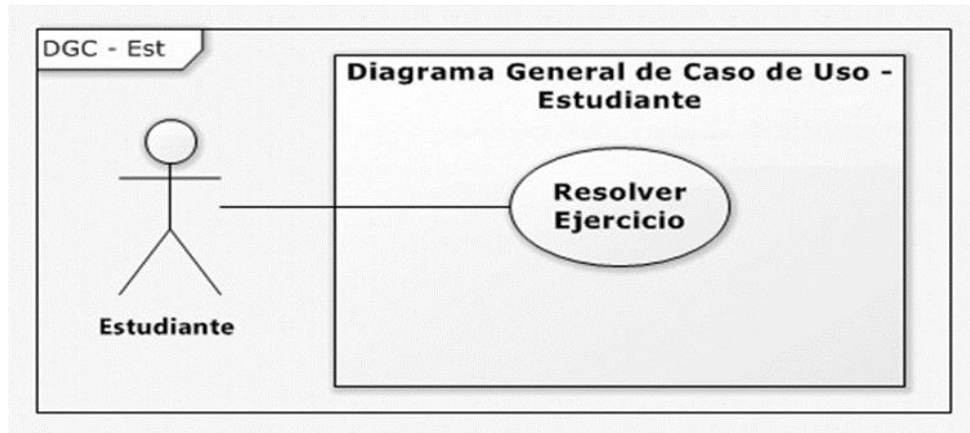
4.7.2 Profesor

Figura 5. Diagrama de Caso de Uso - Profesor



4.7.3 Estudiante

Figura 6. Diagrama de Caso de Uso - Estudiante



4.8 DESCRIPCIÓN DE DIAGRAMAS DE CASOS DE USO

4.8.1 Actualizar Teoría

Figura 7. Diagrama de Caso de Uso – Actualizar Teoría

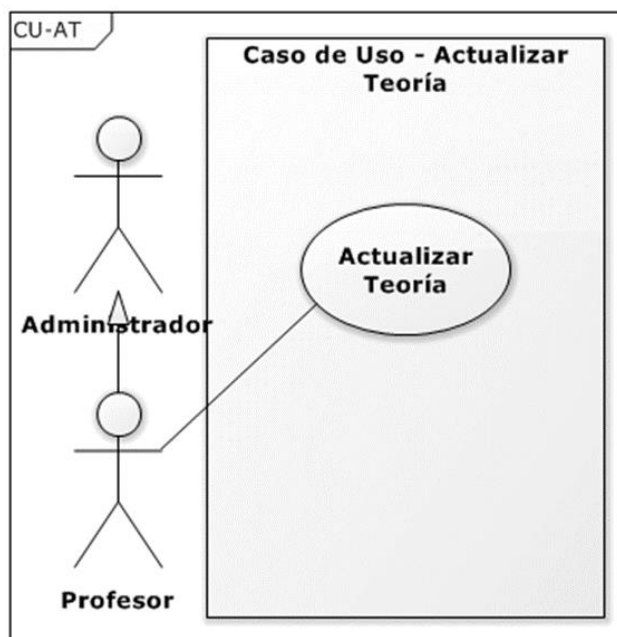


Tabla 14. Descripción de Caso de Uso – Actualizar Teoría

Nombre	Actualizar Teoría del Modelo de Toulmin	ID	CU 001
Descripción			
Permite actualizar la Teoría del Modelo de Toulmin.			
Actores			
Administrador y Profesor.			
Disparadores			
Elegir la opción de Actualizar Teoría del Modelo de Toulmin.			
Precondiciones			
Para poder ingresar a la opción de Actualizar de la sección de Inicio deberá estar autenticado como Administrador o Profesor.			
Flujo Normal			
<ol style="list-style-type: none"> 1. El Administrador o Profesor elige la opción de Actualizar Teoría del Modelo de Toulmin. 2. El sistema muestra el formulario de actualización de datos de la Teoría del Modelo de Toulmin. 3. El Administrador o Profesor diligencia el formulario de actualización de Teoría del Modelo de Toulmin. 4. El sistema comprueba el correcto diligenciamiento de la información ingresada en la base de datos de la Teoría del Modelo de Toulmin. 			
Flujo Alternativo			
El formulario de actualización de Teoría del Modelo de Toulmin no fue diligenciado de forma correcta.			
<ol style="list-style-type: none"> 4.1. El sistema verifica el correcto diligenciamiento de los campos de la Teoría del Modelo de Toulmin. 			
Postcondiciones			
La Teoría del modelo de Toulmin ha sido actualizada en la base de datos.			

4.8.2 Actualizar Ejemplo

Figura 8. Diagrama de Caso de Uso – Actualizar Ejemplo

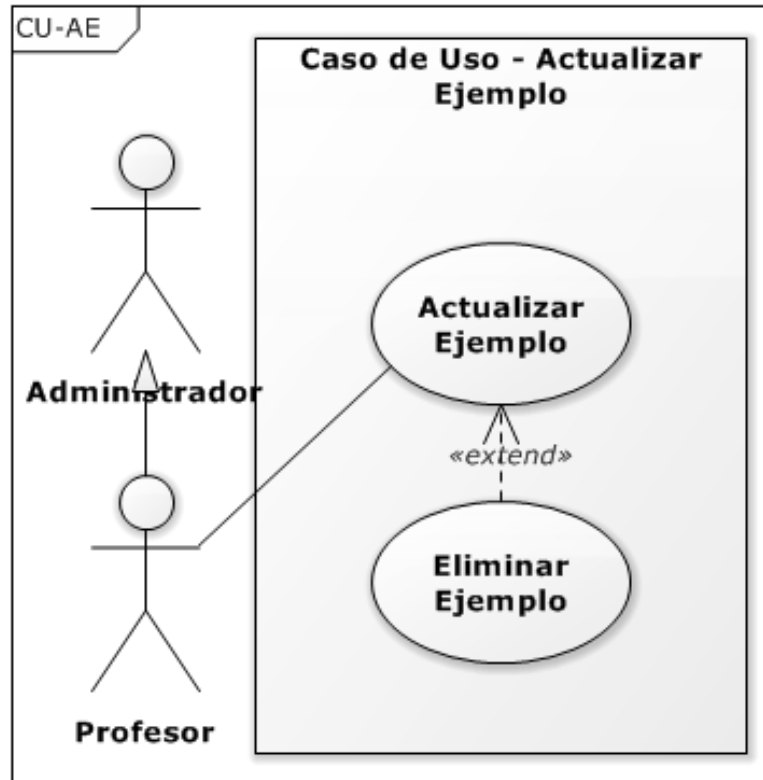


Tabla 15. Descripción de Caso de Uso – Actualizar Ejemplo

Nombre	Actualizar Ejemplo del Modelo de Toulmin	ID	CU 002
Descripción	Permite actualizar un Ejemplo del Modelo de Toulmin.		
Actores	Administrador y Profesor.		
Disparadores	Elegir la opción de Actualizar un Ejemplo del Modelo de Toulmin.		
Precondiciones	Para poder ingresar a la opción de Actualizar de la sección de Ejemplos deberá estar autenticado como Administrador o Profesor.		
Flujo Normal	<ol style="list-style-type: none"> 1. El Administrador o Profesor elige la opción de Actualizar un Ejemplo del Modelo de Toulmin. 2. El sistema muestra un formulario de la actualización de datos de Ejemplos del Modelo de Toulmin. 3. El Administrador o Profesor diligencia el formulario de actualización de datos del Ejemplo del Modelo de Toulmin. 		

4. El sistema comprueba el correcto diligenciamiento de la información ingresada en la base de datos de los Ejemplos del Modelo de Toulmin.
Flujo Alternativo
El formulario de actualización del Ejemplo del Modelo de Toulmin no fue diligenciado de forma correcta.
4.1. El sistema verifica el correcto diligenciamiento de los campos del Ejemplo del Modelo de Toulmin.
Postcondiciones
El Ejemplo del modelo de Toulmin ha sido actualizado en la base de datos.

4.8.3 Actualizar Ejercicio

Figura 9. Diagrama de Caso de Uso – Actualizar Ejercicio

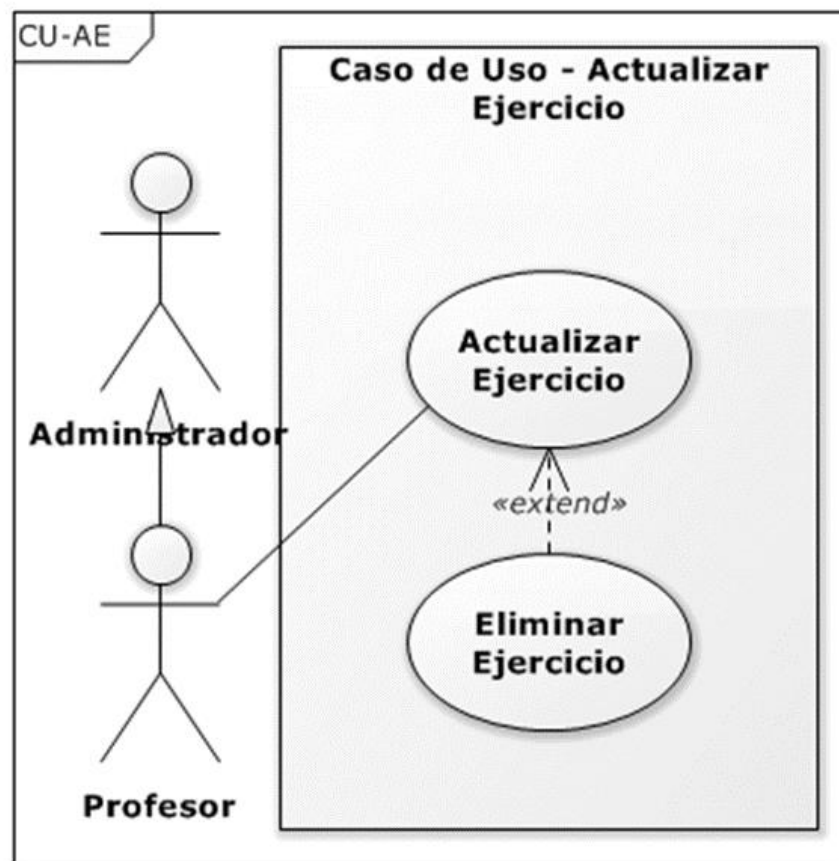


Tabla 16. Descripción de Caso de Uso – Actualizar Ejercicio

Nombre	Actualizar Ejercicio de Modelo de Toulmin	ID	CU 003
Descripción			
Permite actualizar un Ejercicio del Modelo de Toulmin.			
Actores			
Administrador y Profesor.			
Disparadores			
Elegir la opción de Actualizar un Ejercicio del Modelo de Toulmin.			
Precondiciones			
Para poder ingresar a la opción de Actualizar de la sección de Ejercicios deberá estar autenticado como Administrador o Profesor.			
Flujo Normal			
<ol style="list-style-type: none"> 1. El Administrador o Profesor elige la opción de Actualizar un Ejercicio del Modelo de Toulmin. 2. El sistema muestra el formulario de actualización de datos de Ejercicios del Modelo de Toulmin. 3. El Administrador o Profesor diligencia el formulario de actualización de datos del Ejercicio del Modelo de Toulmin. 4. El sistema comprueba el correcto diligenciamiento de la información ingresada en la base de datos de los Ejercicios del Modelo de Toulmin. 			
Flujo Alternativo			
<p>El formulario de actualización del Ejercicio del Modelo de Toulmin no fue diligenciado de forma correcta.</p> <ol style="list-style-type: none"> 4.1. El sistema verifica el correcto diligenciamiento de los campos del Ejercicio del Modelo de Toulmin. 			
Postcondiciones			
El Ejercicio del modelo de Toulmin ha sido actualizado en la base de datos.			

4.8.4 Registrar Ejemplo

Figura 10. Diagrama de Caso de Uso – Registrar Ejemplo

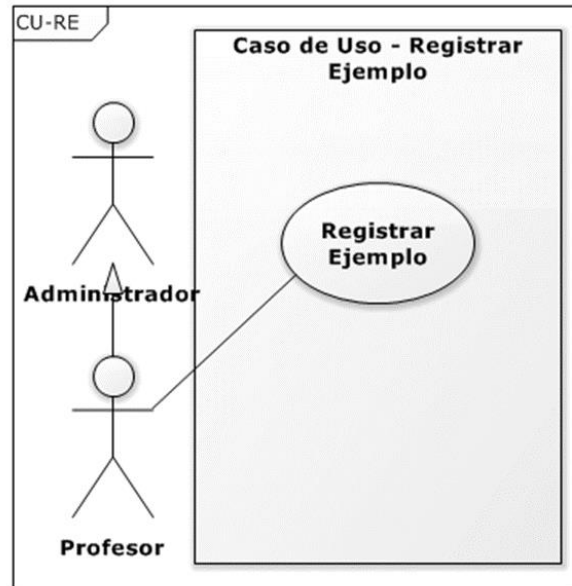


Tabla 17. Descripción de Caso de Uso – Registrar Ejemplo

Nombre	Registrar Ejemplo del Modelo de Toulmin	ID	CU 005
Descripción			
Permite registrar un nuevo Ejemplo del Modelo de Toulmin.			
Actores			
Administrador y Profesor.			
Disparadores			
Elegir la opción de Registrar un Ejemplo del Modelo de Toulmin.			
Precondiciones			
Para poder ingresar a la opción de Registrar de la sección de Ejemplos deberá estar autenticado como Administrador o Profesor.			
Flujo Normal			
<ol style="list-style-type: none"> 1. El Administrador o Profesor elige la opción de Registrar un Ejemplo del Modelo de Toulmin. 2. El sistema muestra el formulario de actualización de datos de un Ejemplo del Modelo de Toulmin. 3. El Administrador o Profesor diligencia el formulario de actualización de datos de Ejemplos del Modelo de Toulmin. 4. El sistema comprueba el correcto diligenciamiento de la información ingresada en la base de datos de los Ejemplos del Modelo de Toulmin. 			
Flujo Alternativo			
El formulario de actualización de Ejemplos del Modelo de Toulmin no fue diligenciado de forma correcta.			
4.1. El sistema verifica el correcto diligenciamiento de los campos del Ejemplo del Modelo de Toulmin.			
Postcondiciones			
El Ejemplo del Modelo de Toulmin ha sido registrado en la base de datos.			

4.8.5 Registrar Ejercicio

Figura 11. Diagrama de Caso de Uso – Registrar Ejercicio

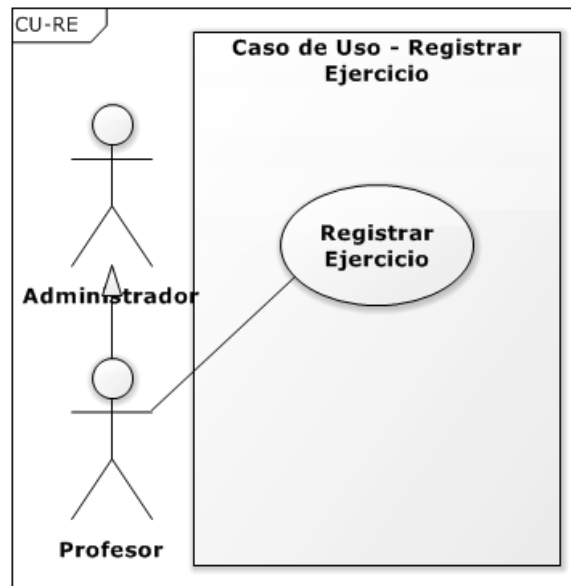


Tabla 18. Descripción de Caso de Uso – Registrar Ejercicio

Nombre	Registrar Ejercicio de Modelo de Toulmin	ID	CU 006
Descripción	Permite registrar un nuevo Ejercicio del Modelo de Toulmin.		
Actores	Administrador y Profesor.		
Disparadores	Elegir la opción de Registrar un Ejercicio del Modelo de Toulmin.		
Precondiciones	Para poder ingresar a la opción de Agregar de la sección de Ejercicios deberá estar autenticado como Administrador o Profesor.		
Flujo Normal	<ol style="list-style-type: none"> 1. El Administrador o Profesor elige la opción de Registrar un Ejercicio del Modelo de Toulmin. 2. El sistema muestra el formulario de actualización de datos de Ejercicios del Modelo de Toulmin. 3. El Administrador o Profesor diligencia el formulario de actualización de datos de un Ejercicio del Modelo de Toulmin. 4. El sistema comprueba el correcto diligenciamiento de la información ingresada en la base de datos de los Ejercicios del Modelo de Toulmin. 		
Flujo Alternativo	<p>El formulario de actualización del Ejercicio del Modelo de Toulmin no fue diligenciado de forma correcta.</p> <ol style="list-style-type: none"> 4.1. El sistema verifica el correcto diligenciamiento de los campos del Ejercicio del Modelo de Toulmin. 		
Postcondiciones	El Ejercicio del Modelo de Toulmin ha sido registrado en la base de datos.		

4.8.6 Resolver Ejercicio

Figura 12. Diagrama de Caso de Uso – Resolver Ejercicio

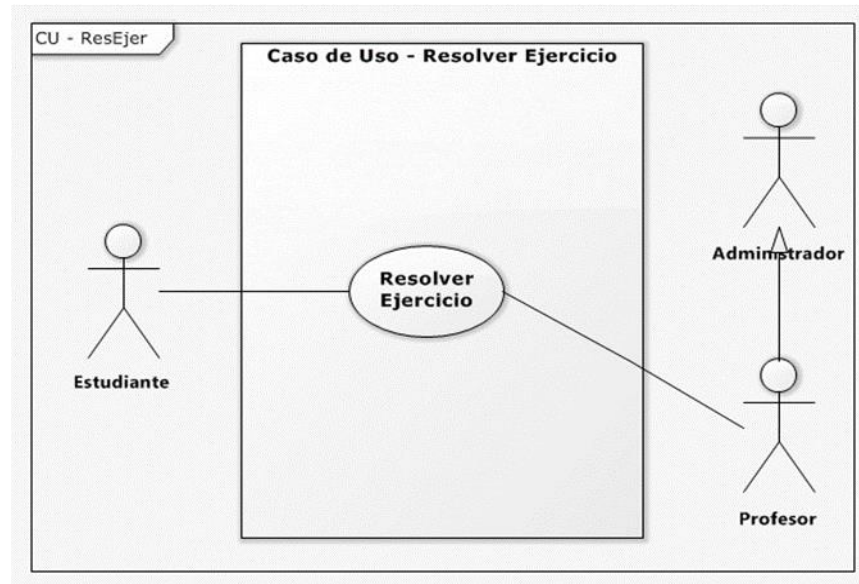


Tabla 19. Descripción de Caso de Uso – Resolver Ejercicio

Nombre	Resolver Ejercicio de Modelo de Toulmin	ID	CU 008
Descripción	Permite resolver un Ejercicio del Modelo de Toulmin.		
Actores	Profesor, Administrador y Estudiante.		
Disparadores	Elegir la opción de Resolver un Ejercicio del Modelo de Toulmin.		
Precondiciones	Para poder ingresar a la opción de Ejercicios deberá estar autenticado como Administrador, Profesor o Estudiante.		
Flujo Normal	<ol style="list-style-type: none"> 1. El usuario elige la opción de Resolver un Ejercicio del Modelo de Toulmin. 2. El sistema muestra el Ejercicio del Modelo de Toulmin. 3. El usuario resuelve el Ejercicio del Modelo de Toulmin. 4. El sistema comprueba el correcto diligenciamiento de la información ingresada en la base de datos de los Ejercicios del Modelo de Toulmin. 5. El sistema muestra el resultado obtenido por el usuario al resolver el Ejercicio del Modelo de Toulmin. 		
Flujo Alternativo	<p>El Ejercicio del Modelo de Toulmin no fue diligenciado de forma correcta.</p> <ol style="list-style-type: none"> 4.1. El sistema verifica el correcto diligenciamiento de los campos del Ejercicio del Modelo de Toulmin. 		
Postcondiciones	El puntaje del Ejercicio realizado queda almacenado en la base de datos.		

5. DEFINICIÓN DEL MODELO

Terminada una vez la fase de análisis de requerimientos. Para la primera iteración se hizo una respectiva reunión entre el director, los desarrolladores y el representante del grupo de filosofía para llegar a un acuerdo sobre la estructura básica del Modelo de Toulmin, con la que se realizó la plantilla para los ejercicios y/o ejemplos. La cual consta de seis categorías: Conclusión, datos, garantías, apoyos, calificador modal y refutación. Además, se llegó al acuerdo sobre correlación existente entre algunas de las partes del modelo representadas en la siguiente plantilla.

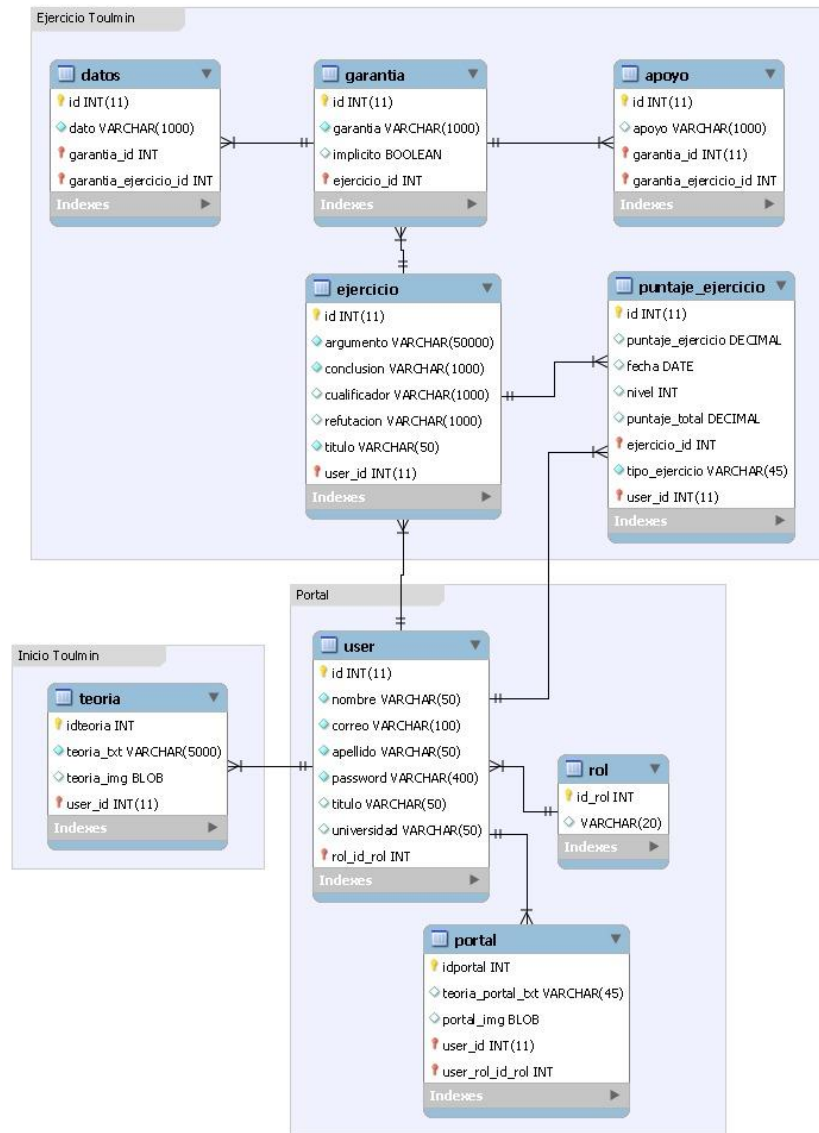
Figura 13. Plantilla de recolección de información

Argumento Argumento escrito de forma completa <div style="border: 1px solid black; height: 20px; width: 100%; margin-top: 5px;"></div>														
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 5px;">Datos</th> <th style="text-align: center; padding: 5px;">Garantías</th> <th style="text-align: center; padding: 5px;">Respaldo o Apoyo</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;">Hechos, motivos, razones o evidencias del caso que sirven para soportar la conclusión</td> <td style="padding: 5px;">Establece cómo los datos sirven de soporte legítimo a la conclusión</td> <td style="padding: 5px;">Estudios científicos, códigos, estadísticas o creencia de una comunidad que respaldan la garantía</td> </tr> <tr> <td style="height: 20px;"></td> <td></td> <td></td> </tr> <tr> <td style="height: 20px;"></td> <td></td> <td></td> </tr> </tbody> </table>	Datos	Garantías	Respaldo o Apoyo	Hechos, motivos, razones o evidencias del caso que sirven para soportar la conclusión	Establece cómo los datos sirven de soporte legítimo a la conclusión	Estudios científicos, códigos, estadísticas o creencia de una comunidad que respaldan la garantía								
Datos	Garantías	Respaldo o Apoyo												
Hechos, motivos, razones o evidencias del caso que sirven para soportar la conclusión	Establece cómo los datos sirven de soporte legítimo a la conclusión	Estudios científicos, códigos, estadísticas o creencia de una comunidad que respaldan la garantía												
Conclusión La tesis que se va a defender, el asunto a debatir, a demostrar o a sostener. El enunciado que se justifica a partir de la garantía y el dato <div style="border: 1px solid black; height: 20px; width: 100%; margin-top: 5px;"></div>														
Refutación o restricción Es la excepción de la conclusión presentada <div style="border: 1px solid black; height: 20px; width: 100%; margin-top: 5px;"></div>														
Cualificador modal o matizador Expresión del grado de certidumbre del enunciador en relación a la conclusión <div style="border: 1px solid black; height: 20px; width: 100%; margin-top: 5px;"></div>														

5.1 DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS

Una vez claro el modelo a trabajar de S. Toulmin, se procede al diseño e implementación de la base de datos, reuniéndose las partes involucradas, profesor y estudiantes encargados del desarrollo y se acordó que será relacional. En la siguiente imagen se muestra la base de datos completa implementada en el desarrollo.

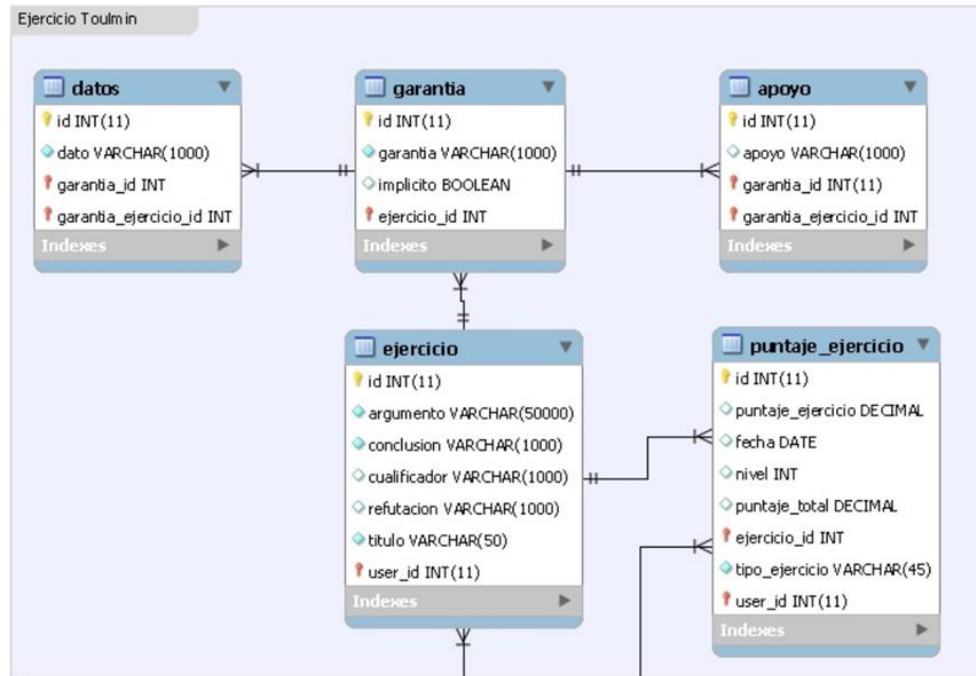
Figura 14. Base de datos de la aplicación



La base de datos se puede dividir en tres grandes cuerpos, los cuales son:

1. Ejercicio de Toulmin: Encargada del almacenamiento de las categorías correspondientes al modelo, así como su respectivo puntaje al ejercicio realizado.

Figura 15. Base de datos: Encargada del Modelo de Toulmin



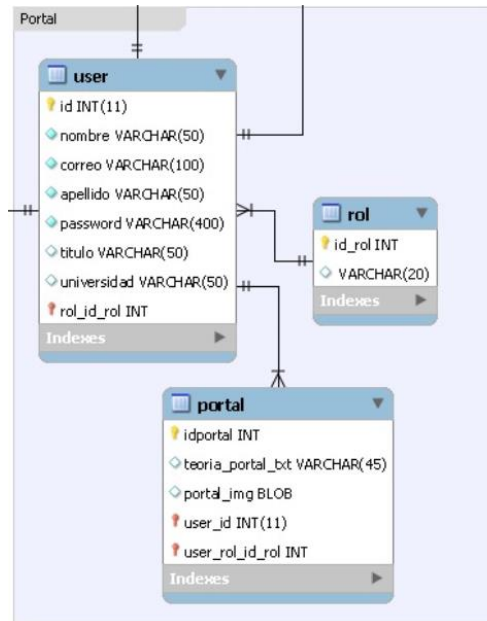
2. Inicio Toulmin: Encargada del almacenamiento del inicio de la aplicación, que consta de una imagen y un texto explicativo sobre el modelo.

Figura 16. Base de datos: Encargada del Inicio de la aplicación



3. Portal: Encargada principalmente de la relación con el usuario que viene desde el portal Filoen directamente y que nos ofrece principalmente el rol del este.

Figura 17. Base de datos: Encargada de la información de los usuarios y sus relaciones



5.2 DESARROLLO

Según la metodología escogida y propuesta inicialmente (modelo de prototipos), se procede entonces a realizar las tres iteraciones.

1. **Primera iteración:** Se procede a realizar un primer desarrollo (versión de prueba) funcional, en el cual los usuarios finales puedan tener un primer acercamiento y de tal manera se obtiene una idea más clara de lo que el usuario busca.

Figura 18. Primera iteración: Vista del Home

Teoría

El modelo de Toulmin (1958), profundizado en Toulmin, Rieke, and Janik (1984), se relaciona con las reglas de una argumentación en pasos que pueden ser precisados en cualquier tipo de disciplina o espacio abierto a la disertación, al debate. Mediante este modelo, los docentes pueden motivar a los estudiantes a encontrar la evidencia que fundamenta una aserción. Se aprende que la excelencia de una argumentación depende de un conjunto de relaciones que pueden ser precisadas y examinadas y que el lenguaje de la razón esta presente en todo tipo de discurso.

Toulmin (1958) cree que las argumentaciones cotidianas no siguen el clasico modelo riguroso del silogismo y crea uno adecuado para analizar cualquier tipo de argumentación en el marco de los discursos sociales: conversación, periodico, television, radio, prensa escrita, entrevista, interacción docente alumno, medico-paciente, abogado-cliente. Considera que un argumento es una estructura compleja de datos que involucra un movimiento que parte de una evidencia (grounds) y llega al establecimiento de una aserción (tesis, causa). El movimiento de la evidencia a la aserción (claim) es la mayor prueba de que la línea argumental se ha realizado con efectividad. La garantía permite la conexión.

Aserción: los resultados de las elecciones, posiblemente, no serán confiables.
 Datos: Los partidos políticos tradicionales han hecho trampa en todas las elecciones.
 Garantía: Si antes han actuado con trampa, probablemente siempre la volverán a cometer. (Creencia común).
 Los otros tres pasos del modelo son respaldo, cualificador modal y reserva. Así la garantía anterior tiene un respaldo en estudios realizados por expertos sobre el comportamiento de los políticos en las elecciones con base en datos estadísticos, en testimonios orales, historias de vida, entre otros. El cualificador modal indica el grado de fuerza o de probabilidad de la aserción. La reserva habla de las posibles objeciones que se le puedan formular.



Figura 19. Primera iteración: Vista Agregar Ejercicios

Ejercicio tipo 1: vista profesor

Argumento: Petersen es Sueco, y ya que los Suecos en general no son católicos, con menos que el 2% siendo católicos, es casi seguro de que no es católico, a menos que es uno del 2% que es católico

Rellenar los espacios correspondientes con las partes del argumento expuesto arriba:

Datos:	Los Suecos en general no son católicos.
Garantía:	No es católico.
Apoyo:	Es uno del 2% de católicos.
conclusion:	Petersen es Sueco.
matizador:	Es casi seguro.
refutación:	Menos que el 2% de los suecos son católicos.

enviar

Figura 20. Primera iteración: Vista Desarrollar Ejercicios

Ejercicio 1: vista estudiante

Petersen es Sueco, y ya que los Suecos en general no son católicos, con menos que el 2% siendo católicos, es casi seguro de que no es católico, a menos que es uno del 2% que es católico

dato	Es casi seguro.
garantía	Los Suecos en general no son católicos.
apoyo	Es uno del 2% de católicos.
conclusion	Petersen es Sueco.
matizador	Menos que el 2% de los suecos son católicos.
refutación	No es católico.

revisar respuestas

2. **Segunda iteración:** Con base a los resultados obtenidos durante la primera iteración se procede entonces a desarrollar una mejor interfaz gráfica para una mejor usabilidad a la aplicación. Donde el usuario ya puede interactuar de una mejor manera con la parte administrativa del sistema.

Figura 21. Segunda iteración: Vista Home

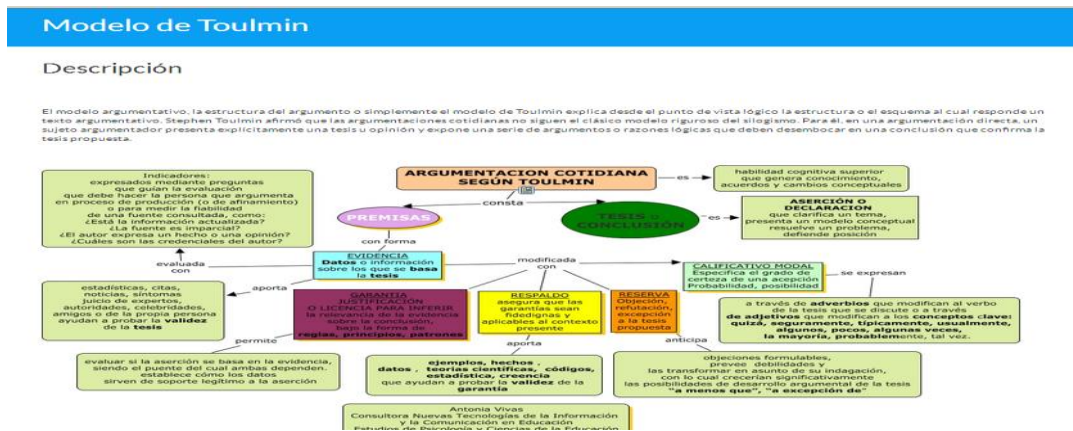


Figura 22. Segunda iteración: Vista de la Teoría

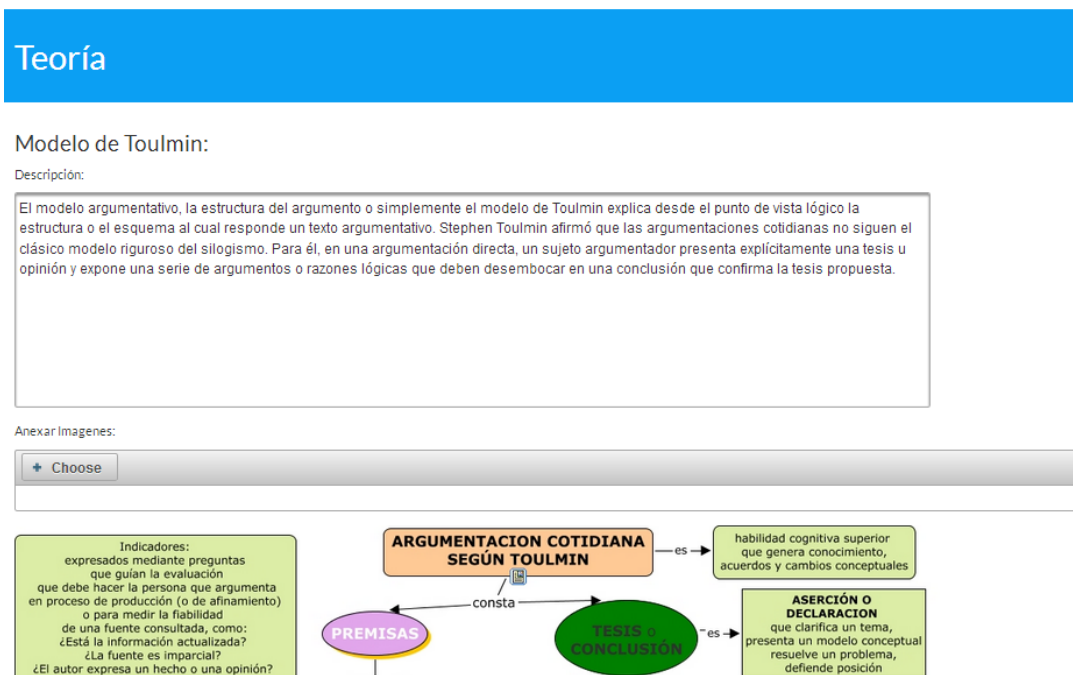


Figura 23. Segunda iteración: Vista de los ejercicios



Figura 24. Segunda iteración: Visualización Modificar Ejercicio

Ejercicio

Modelo de Toulmin:

Titulo

Argumento

Datos Garantía Apoyo

Lista de Ejercicios			
Datos	Argumentos	Apoyo	
No records found.			

Conclusion

Refutación

Cualificador Moral

3. **Tercera iteración:** Se proceden a realizar ajustes al modelo de los ejercicios para su versión final y se hace la conexión con la base de datos de los usuarios del portal para su respectiva interoperabilidad según su rol. Obteniendo así una versión final más acertada a lo que busca el usuario desde un principio y en el cual ya obtiene un resultado de su desempeño.

Figura 25. Tercera iteración: Vista actualizar teoría

Teoría

Modelo de Toulmin:

Descripción:

Antes de discutir en detalle el modelo de Toulmin, resulta pertinente introducir las nociones de argumento y argumentación que se emplearán en esta revisión: un argumento se refiere a los discursos que un estudiante o un grupo de estudiantes producen cuando deben articular o justificar sus conclusiones o explicaciones, mientras que la argumentación alude al proceso de elaboración de esos discursos (OSBORNE; ERDURAN; SIMON, 2004).

Uno de los aspectos más interesantes del modelo de Toulmin – designado habitualmente como TAP por las siglas en inglés de Toulmin's argument pattern –es que ofrece un potente enfoque para estudiar lo que el filósofo británico denomina argumentos sustantivos, es decir, aquellos que deben ser examinados atendiendo a su contenido, lo cual marca una profunda diferencia con la tradición aristotélica, que se interesa únicamente por la forma o estructura de un argumento.

Es entonces cuando Toulmin propone que las argumentaciones cotidianas no siguen el clásico modelo riguroso del silogismo y crea uno adecuado para analizar cualquier tipo de argumentación en el marco de los discursos sociales: conversación, periódico, televisión, radio, prensa escrita, entrevista, interacción docente alumno, médico-paciente,

Anexar Imágenes:

```

graph TD
    D["D: Péndulo de Foucault.  
El plano de oscilación parece girar"] -- "Por lo tanto" --> Q["Q: Probablemente,  
presumiblemente,  
algunas veces, etc."]
    D -- "Porque" --> G["G: El plano de oscilación  
no se mueve, tiene que  
ser la Tierra la que gira"]
    Q --> C["C: La tierra gira sobre su  
propio eje."]
    C -- "A menos que" --> R["R: Existe una fuerza no  
considerada (mecánica,  
magnética, etc.) que"]
  
```


Figura 26. Tercera iteración: Vista de los Ejemplos

Ejemplos

Lista de Ejemplos		
Id	Titulo	
15	elecciones	<input type="button" value="Ver"/>
16	el arte y la publicidad	<input type="button" value="Ver"/>
17	Herencia	<input type="button" value="Ver"/>
18	los ciclistas colombianos	<input type="button" value="Ver"/>
19	Petersen el sueco	<input type="button" value="Ver"/>
20	Rick	<input type="button" value="Ver"/>
21	Juan mató a Manuel?	<input type="button" value="Ver"/>
22	juan el ecuatoriano	<input type="button" value="Ver"/>
23	Barcelona es mejor	<input type="button" value="Ver"/>
24	Real Madrid es mejor	<input type="button" value="Ver"/>

(1 of 1) 1 10

Figura 27. Tercera iteración: Vista de un ejemplo

[Home](#)
[Teoría](#)
[Ejercicios](#)
[Ejemplos](#)
[Solucionar Ejercicios](#)


Ejemplo -> elecciones

Modelo de Toulmin:

Argumento:
 Los resultados de las elecciones, posiblemente, no serán confiables. Y es que los partidos políticos tradicionales han hecho trampa en todas las elecciones. Si antes han actuado con trampa, probablemente siempre la volverán a cometer. En un informe de Portillo (un experto), concluye que los países acostumbrados al fraude electoral, tratan siempre de perpetuar sus prácticas. A menos que todos y cada uno de los partidos políticos tengan una representación en los escrutinios y que, además, una comisión de ética vigile que los grupos minoritarios no vendan sus votos.

Datos

Los partidos políticos tradicionales han hecho trampa en todas las elecciones.

Respaldo

Portillo (un experto) concluye que los países acostumbrados al fraude electoral, tratan siempre de perpetuar sus prácticas.

Garantía

si antes han actuado con trampa, probablemente siempre la volverán a cometer. (Creencia común).

Cualificador

posiblemente.


Conclusión

los resultados de las elecciones, posiblemente, no serán confiables.

Refutación

A menos que todos y cada uno de los partidos políticos tengan una representación en los escrutinios y que, además, una comisión de ética vigile que los grupos minoritarios no vendan sus votos.

Figura 28. Tercera iteración: Vista modificar ejercicio

[Home](#)
[Teoría](#)
[Ejercicios](#)
[Ejemplos](#)
[Solucionar Ejercicios](#)


Ejercicio -> Herencia

[Finalizar](#)

A partir del argumento dado, identifique cada uno de las partes del Modelo de Toulmin. Para esto deberá arrastrar cada recuadro a su respectiva parte del modelo

Argumento:
 X debería recibir la herencia, ya que X es el único hijo de Y, quien falleció sin dejar testamento. Los hijos suceden a los padres cuando éstos han fallecido sin dejar testamento, como está establecido en el artículo 930 del Código Civil. Salvo que X incurra en causa de desheredación

Disponibles

Ubícame

Debería

Ubícame

El artículo 930 del Código Civil

Ubícame

X es el único hijo de Y, quien falleció sin dejar testamento

Ubícame

Salvo que X incurra en causa de desheredación.

Ubícame

Los hijos suceden a los padres cuando éstos han fallecido sin dejar testamento

Ubícame

Y debería recibir la herencia

Datos

Ingrese su respuesta aquí

Respaldo

Ingrese su respuesta aquí

Garantía

Ingrese su respuesta aquí

Cualificador


Ingrese su respuesta aquí

Conclusión

Ingrese su respuesta aquí

Refutación

Ingrese su respuesta aquí



Home | Ejercicios | Teoría | Solucionar Ejercicios
Copyright © 2024. Todos los derechos reservados. WebTema.com

Figura 29. Tercera iteración: Visualización realizar Ejercicio

Ejercicio ->

Finalizar

A partir del argumento dado, identifique cada una de las partes del Modelo de Toulmin. Para esto deberá arrastrar cada recuadro a su respectiva parte del modelo

Argumento:

los resultados de las elecciones, posiblemente, no serán confiables. Y es que los partidos políticos tradicionales han hecho trampa en todas las elecciones. si antes han actuado con trampa, probablemente siempre la volverán a cometer. En un informe de Portillo (un experto), concluye que los países acostumbrados al fraude electoral, tratan siempre de perpetuar sus prácticas. : A menos que todos y cada uno de los partidos políticos tengan una representación en los escrutinios y que, además, una comisión de ética vigile que los grupos minoritarios no vendan sus votos.

Disponibles

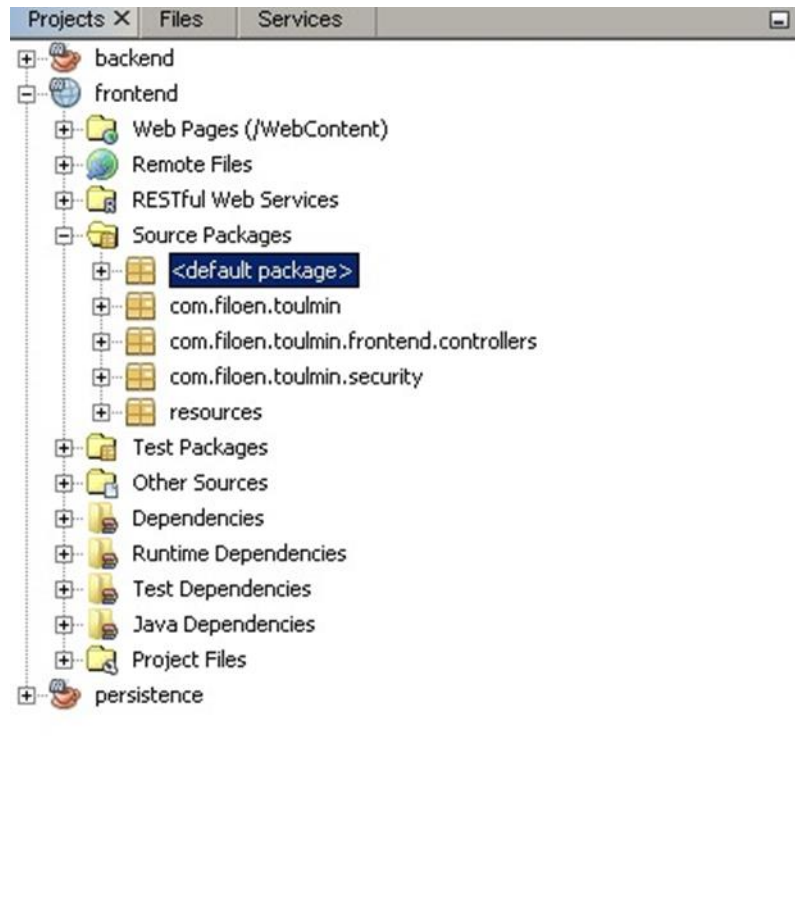
Ubicame Portillo (un experto) concluye que los países acostumbrados al fraude electoral, tratan siempre de perpetuar sus prácticas.	Ubicame posiblemente.	Ubicame los partidos políticos tradicionales han hecho trampa en todas las elecciones.	Ubicame si antes han actuado con trampa, probablemente siempre la volverán a cometer. (Creencia común).
Ubicame los resultados de las elecciones, posiblemente, no serán confiables.	Ubicame A menos que todos y cada uno de los partidos políticos tengan una representación en los escrutinios y que, además, una comisión de ética vigile que los grupos minoritarios no vendan sus votos.		

Datos Ingrese su respuesta aquí	Respaldo Ingrese su respuesta aquí	Garantía Ingrese su respuesta aquí	Cualificador Ingrese su respuesta aquí	Conclusión Ingrese su respuesta aquí
			Refutación Ingrese su respuesta aquí	

IMPLEMENTACIÓN DEL DISEÑO DE CLASES

Frontend contiene la parte visual de la aplicación, con lo que el usuario interactúa: los xhtml y los controlers de Java Server Faces. Persistence, las clases y objetos que tienen que ver con la información en la base de datos, además de las entidades y los repositorios de spring con que se accede a base de datos. Y en el backend, lo que se encarga de las interacciones que hay entre el servidor y la base de datos.

Figura 30. Visualización del backend, frontend y el persistence



5.3 PRUEBAS DE VERIFICACION

Se verificó que la aplicación web cumpliera con los requisitos y los casos de uso establecidos durante la fase de análisis, donde se llevó a cabo las diferentes pruebas a cada uno de los módulos, donde se ingresaron datos necesarios para ejecutar cada una de las funcionalidades. Dichas pruebas se realizaron durante cada fase o iteración realizada, esto para su respectivo análisis, evaluación y realización de ajustes necesarios hasta llegar al producto final deseado.

5.3.1 Primera Iteración

Tabla 20. Resultados de las pruebas aplicadas la herramienta (iteración 1).

CASOS DE PRUEBA	RESULTADO FECHA: 25/03/2018
Existencia de la Interfaz de Home	Exitoso
Existencia de la Interfaz de Teoría	Exitoso
Existencia de la Interfaz de Ejercicios	Exitoso
Existencia de la Interfaz de Ejemplos	Exitoso
Existencia de la Interfaz de Solucionar Ejercicios	Exitoso
Validación de datos para guardar texto de Teoría	Exitoso
Validación de datos para agregar imagen de Teoría	Exitoso
Existencia de la Interfaz para agregar nuevo ejercicio	Exitoso
Validación de datos para agregar nuevo Ejercicio	Sin Éxito
Visualizar ejemplo	Sin Éxito
Resolver ejercicio	Sin Éxito
Visualizar resultados al Responder ejercicio	Sin Éxito

5.3.2 Segunda Iteración

Tabla 21. Resultados de las pruebas aplicadas la herramienta (iteración 2)

CASOS DE PRUEBA	RESULTADO FECHA: 11/07/2018
Existencia de la Interfaz de Home	Exitoso
Existencia de la Interfaz de Teoría	Exitoso
Existencia de la Interfaz de Ejercicios	Exitoso
Existencia de la Interfaz de Ejemplos	Exitoso
Existencia de la Interfaz de Solucionar Ejercicios	Exitoso
Validación de datos para guardar texto de Teoría	Exitoso
Validación de datos para agregar imagen de Teoría	Exitoso
Existencia de la Interfaz para agregar nuevo ejercicio	Exitoso
Validación de datos para agregar nuevo Ejercicio	Exitoso
Eliminar ejercicio	Exitoso
Validación de datos para modificar ejercicio	Exitoso
Visualizar ejemplo	Exitoso
Responder ejercicio	Exitoso
Visualizar resultados al Responder ejercicio	Exitoso
Buscar ejercicios por título	Sin Éxito
Buscar ejemplos por título	Sin Éxito
Buscar ejercicios para solucionar por título	Sin Éxito

5.3.3 Prototipo Final

Tabla 22. Resultados de las pruebas aplicadas a la interfaz de Portal (Producto Final).

CASOS DE PRUEBA	RESULTADO FECHA: 12/04/2019
Existencia de la Interfaz de Inicio	Exitoso
Existencia de la Interfaz de Registrarse	Exitoso
Existencia de la Interfaz de Iniciar Sesión	Exitoso

Tabla 23. Resultados de las pruebas aplicadas a la interfaz de Iniciar de Sesión (Producto Final).

CASOS DE PRUEBA	RESULTADO FECHA: 12/04/2019
Existencia de la Interfaz de Iniciar Sesión.	Exitoso
Ingreso de los diferentes usuarios con su respectiva sesión.	Exitoso

Tabla 24. Resultados de las pruebas aplicadas a la interfaz de Registrarse (Producto Final).

CASOS DE PRUEBA	RESULTADO FECHA: 12/04/2019
Existencia de la Interfaz de Registrarse.	Exitoso
Validación de datos para el registro de un usuario.	Exitoso

Tabla 25. Resultados de las pruebas aplicadas la herramienta para el rol de Profesor (Producto Final).

CASOS DE PRUEBA	RESULTADO FECHA: 12/04/2019
Existencia de la Interfaz de Home	Exitoso
Existencia de la Interfaz de Teoría	Exitoso
Existencia de la Interfaz de Ejercicios	Exitoso
Existencia de la Interfaz de Ejemplos	Exitoso
Existencia de la Interfaz de Solucionar Ejercicios	Exitoso
Validación de datos para guardar texto de Teoría	Exitoso
Validación de datos para agregar imagen de Teoría	Exitoso
Existencia de la Interfaz para agregar nuevo ejercicio	Exitoso
Validación de datos para agregar nuevo Ejercicio	Exitoso
Eliminar ejercicio	Exitoso
Validación de datos para modificar ejercicio	Exitoso
Visualizar ejemplo	Exitoso
Responder ejercicio	Exitoso

Visualizar resultados al Responder ejercicio	Exitoso
Buscar ejercicios por título	Exitoso
Buscar ejemplos por título	Exitoso
Buscar ejercicios para solucionar por título	Exitoso
Cerrar sesión	Exitoso

Tabla 26. Resultados de las pruebas aplicadas la herramienta para el rol de Estudiante (Producto Final).

CASOS DE PRUEBA	RESULTADO FECHA: 12/04/2019
Existencia de la Interfaz de Home	Exitoso
Existencia de la Interfaz de Ejemplos	Exitoso
Existencia de la Interfaz de Solucionar Ejercicios	Exitoso
Responder ejercicio	Exitoso
Visualizar ejemplo	Exitoso
Visualizar resultados al responder ejercicio	Exitoso
Buscar ejercicios por título	Exitoso
Buscar ejemplos por título	Exitoso
Buscar ejercicios para solucionar por título	Exitoso
Cerrar sesión	Exitoso

5.4 DESCRIPCION DE PRUEBAS DE VERIFICACIÓN PARA EL PRODUCTO FINAL

5.4.1 Caso de Prueba: 01

Título: Existencia de la interfaz de inicio.

Fecha: 12/04/2019

Tabla 27. Caso de prueba 01

PASO	VALOR ESPERADO	VALOR OBTENIDO
Ingresar la dirección url del portal web: http://www.xenxebot.com:8080/elearning/home.xhtml	Vista de la interfaz del inicio de portal	Se obtuvo el resultado esperado.

5.4.2 Caso de Prueba: 02

Título: Registrar usuario.

Fecha: 12/04/2019

Precondición: Caso de Prueba 01

Tabla 28. Caso de prueba 02

PASO	VALOR ESPERADO	VALOR OBTENIDO
Ingresar al menú Registrarse, completar todos los campos de registro cumpliendo con las siguientes condiciones: Universidad, Título, Nombre y Apellido, deben ser solo letras; el correo electrónico debe seguir la estructura "xxx@xxx.xxx"; la contraseña debe tener al menos 8 caracteres; el rol debe tener <i>seleccionada una opción</i> .	Es redirigido a la interfaz inicial del portal con su sesión iniciada, por lo que al ingresar al menú de Modelo de Toulmin podrá visualizar las opciones disponibles para el rol correspondiente a su usuario (para el rol Estudiante se podrán visualizar en el menú las opciones: Home, ejemplos y desarrollar ejercicio; para el rol Profesor: Home, teoría, ejercicios, ejemplos y desarrollar ejercicio)	Se obtuvo el resultado esperado.

5.4.3 Caso de Prueba: 03

Título: iniciar sesión.

Fecha: 12/04/2019

Precondición: Caso de Prueba 01, 02.

Tabla 29. Caso de prueba 03

PASO	VALOR ESPERADO	VALOR OBTENIDO
Ingresar al menú Iniciar sesión, llenar los campos Correo Electrónico y contraseña.	Es redirigido a la interfaz inicial del portal con su sesión iniciada, por lo que al ingresar al menú de Modelo de Toulmin podrá visualizar las opciones disponibles para el rol correspondiente a su usuario (para el rol Estudiante se podrán visualizar en el menú las opciones: Home, ejemplos y desarrollar ejercicio; para el rol Profesor: Home, teoría, ejercicios, ejemplos y desarrollar ejercicio)	Se obtuvo el resultado esperado.

5.4.4 Caso de Prueba: 04

Título: Modificar texto de Home.

Fecha: 12/04/2019

Precondición: Caso de Prueba 03.

Tabla 30. Caso de prueba 04

PASO	VALOR ESPERADO	VALOR OBTENIDO
Ingresar al menú Teoría (Solo posible desde el rol de profesor), ingresar texto en el recuadro correspondiente a Descripción, presionar el botón Guardar.	Se actualiza el texto correspondiente a Descripción en Home.	Se obtuvo el resultado esperado.

5.4.5 Caso de Prueba: 05

Título: Modificar la imagen de Home.

Fecha: 12/04/2019

Precondición: Caso de Prueba 03

Tabla 31. Caso de prueba 05

PASO	VALOR ESPERADO	VALOR OBTENIDO
Ingresar al menú Teoría (Solo posible desde el rol de profesor), presionar el botón Agregar Imagen, seleccionar una imagen, presionar abrir.	La imagen se cargará en Teoría y se actualizará inmediatamente en Home.	Se obtuvo el resultado esperado.

5.4.6 Caso de Prueba: 06

Título: Agregar nuevo ejercicio.

Fecha: 12/04/2019

Precondición: Caso de Prueba 03

Tabla 32. Caso de prueba 06

PASO	VALOR ESPERADO	VALOR OBTENIDO
Ingresar al menú Ejercicios (Solo posible desde el rol de profesor), presionar el botón Nuevo, completar todos los campos del formulario (los campos correspondientes a datos, respaldo y garantía se ingresan primero a la tabla "Datos – Resaldos -Garantías" mediante el botón Adicionar/Modificar), presionar el botón Guardar.	Es redirigido a la interfaz de ejercicios, donde aparece el título del ejercicio recientemente agregado con su respectivo número id dentro de la Lista de Ejercicios.	Se obtuvo el resultado esperado

5.4.7 Caso de Prueba: 07

Título: eliminar ejercicio.

Fecha: 12/04/2019

Precondición: Caso de Prueba 03.

Tabla 33. Caso de prueba 07

PASO	VALOR ESPERADO	VALOR OBTENIDO
Ingresar al menú Ejercicios (Solo posible desde el rol de profesor), ubicar el ejercicio específico (puede guiarse por la id del mismo o ingresando palabras que contenga el título del ejercicio en cuestión en el campo Título), presionar el botón eliminar correspondiente al ejercicio, confirmar que se desea eliminar este ejercicio seleccionando la opción aceptar de la ventana emergente.	La Tabla de Ejercicios se actualiza, donde sin el ejercicio eliminado.	Se obtuvo el resultado esperado

5.4.8 Caso de Prueba: 08

Título: Visualizar ejemplo.

Fecha: 12/04/2019

Precondición: Caso de Prueba 03, 06.

Tabla 34. Caso de prueba 08

PASO	VALOR ESPERADO	VALOR OBTENIDO
Ingresar al menú Ejemplos, ubicar el ejemplo específico (puede guiarse por la id del mismo o ingresando palabras que contenga el título del ejemplo en cuestión en el campo Título), presionar el botón Ver.	Se redirige la interfaz de visualización de ejemplos, donde se puede apreciar el ejemplo seleccionado.	Se obtuvo el resultado esperado

5.4.9 Caso de Prueba: 09

Título: Solucionar ejercicio.

Fecha: 12/04/2019

Precondición: Caso de Prueba 03

Tabla 35. Caso de prueba 09

PASO	VALOR ESPERADO	VALOR OBTENIDO
Ingresar al menú Solucionar Ejercicios, ubicar el ejercicio específico (puede guiarse por la id del mismo o ingresando palabras que contenga el título del ejercicio en cuestión en el campo Título), presionar el botón solucionar, Arrastrar los cuadros desde el recuadro de Disponibles hasta los recuadros que indican las partes del argumento, presionar el botón finalizar.	Se redirige a la interfaz de Resultados, donde se indica cuantas partes del argumento fueron respondidas de forma correcta, un botón con la opción de responder otro ejercicio, y se visualiza el argumento completo, seguido de las respuestas ubicadas anteriormente en el esquema de la estructura argumentativa de Toulmin, resaltando en verde las partes correctas y en rojo las incorrectas (para cuando hay errores en las respuestas), a continuación se muestra el esquema de la estructura argumentativa con el ejercicio distribuido correctamente.	Se obtuvo el resultado esperado

6. CONCLUSIONES

El proyecto “Aplicación web de aprendizaje para estructuras argumentativas” muestra la importancia de la ingeniería de sistemas en el ámbito educativo, al automatizar el modelo argumentativo de Toulmin, mediante una plataforma web donde los profesores pueden plantear ejercicios y ejemplos, para que posteriormente sean estudiados y resueltos por los estudiantes desde sus casas, o en su defecto, cualquier lugar con un computador y acceso a internet.

El automatizar la estructura argumentativa garantiza que el estudiante pueda poner a prueba sus conocimientos y capacidades para identificar las partes del argumento con ejercicios y ejemplos prediseñados, sin verse limitado por la disponibilidad y presencia de un tutor.

BIBLIOGRAFÍA

FONTELA, Alvaro. ¿Qué es Bootstrap? [en línea]. <<https://raiolanetworks.es/blog/que-es-bootstrap/>> [citado en 1 de Marzo de 2016].

GAMBOA SARMIENTO, Sonia. Formación y argumentación: automatización de procesos argumentativos. Bucaramanga, 2011. Tesis doctoral (Doctorado Interinstitucional en Educación). Universidad Pedagógica Nacional.

GRUPO FILOSOFÍA Y ENSEÑANZA DE LA FILOSOFÍA. Presentación [en línea] <<http://www.filosofiayensenanza.org/inicio/index.php/grupo/presentacion>>.

IEL, Dan. Usando íconos sin imágenes con Font Awesome [en línea]. <<http://maquetando.com/recursos/fonts/usando-iconos-sin-imagenes-con-font-awesome/>> [citado en 1 de Marzo de 2016].

LAWRENCE, Shari. Ingeniería del software: Teoría y práctica. Pearson Educación, 2002. p. 759. ISBN 9879460715.

MEZA, Paulina. Aproximación al modelo argumentativo de Stephen Toulmin mediante su aplicación a cartas de opinión. En: Simpósio Internacional de Estudos de Gêneros Textuais (V : 2009 : Caxias do Sul). ISSN 1808-7655.

PINOCHET, Jorge. El modelo argumentativo de Toulmin y la educación en ciencias: una revisión argumentada. En: Ciência & Educação (Bauru). Vol. 21, no. 2, p. 307-327.

REAL ACADEMIA ESPAÑOLA. Diccionario de la lengua española. En: Argumento [en línea]. 23 Ed. (2014). <<http://dle.rae.es/?id=3YKtkpX&o=h>>.

TOULMIN, Stephen. Los usos de la argumentación. Traducción de María Morras y Victoria Pineda. 1 ed. Barcelona. Ediciones Península, 2007. 331 p. ISBN 978-84-8307-765-8.

VARGAS, Germán; GAMBOA, Sonia y REEDER, Harry. La humanización como formación: La filosofía y la enseñanza de la filosofía en la condición postmoderna. 1 ed. Bogotá. San Pablo, 2008. 214 p. ISBN 958-692-175.

WIKIPEDIA. Enterprise JavaBeans [en línea]. <https://es.wikipedia.org/wiki/Enterprise_JavaBeans> [citado en 1 de Marzo de 2016].

WIKIPEDIA. Java EE [en línea]. <https://es.wikipedia.org/wiki/Java_EE> [citado en 1 de Marzo de 2016].

WIKIPEDIA. Java Persistence API [en línea]. <https://es.wikipedia.org/wiki/Java_Persistence_API> [citado en 1 de Marzo de 2016].

WIKIPEDIA. JavaServer Faces [en línea]. <https://es.wikipedia.org/wiki/JavaServer_Faces> [citado en 1 de Marzo de 2016].

WIKIPEDIA. MySQL [en línea]. <<https://es.wikipedia.org/wiki/MySQL>> [citado en 1 de Marzo de 2016].

WIKIPEDIA. Netbeans [en línea]. <<https://es.wikipedia.org/wiki/NetBeans>> [citado en 1 de Marzo de 2016].