



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA (ICAI)  
MÁSTER EN INGENIERÍA DE TELECOMUNICACIÓN

# **DISEÑO Y DESARROLLO DE UNA PLATAFORMA DE SOPORTE INTELIGENTE VIRTUAL PARA APRENDIZAJE DE INGLÉS**

Autor: Elena Fernández García  
Director: Miguel Ángel Sanz Bobi

**Madrid**  
Junio 2017

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título Diseño y desarrollo de una plataforma de soporte inteligente virtual para aprendizaje en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el curso académico (2016/17) es de mi autoría, original e inédito y no ha sido presentado con anterioridad a otros efectos.  
El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido tomada de otros documentos está debidamente referenciada.

Fdo.: Elena Fernández García

Fecha: 28/ 06/ 2017

Autorizada la entrega del proyecto  
EL DIRECTOR DEL PROYECTO

Fdo.: Miguel Ángel Sanz Bobi

Fecha: ...../ ...../ .....

Vº Bº del Coordinador de Proyectos

Fdo.: David Contreras Bárcena

Fecha: ...../ ...../ .....



## **AUTORIZACIÓN PARA LA DIGITALIZACIÓN, DEPÓSITO Y DIVULGACIÓN EN RED DE PROYECTOS FIN DE GRADO, FIN DE MÁSTER, TESIS O MEMORIAS DE BACHILLERATO**

### ***1º. Declaración de la autoría y acreditación de la misma.***

El autor D. Elena Fernández García DECLARA ser el titular de los derechos de propiedad intelectual de la obra: Diseño y desarrollo de una plataforma de soporte inteligente virtual para aprendizaje de inglés, que ésta es una obra original, y que ostenta la condición de autor en el sentido que otorga la Ley de Propiedad Intelectual.

### ***2º. Objeto y fines de la cesión.***

Con el fin de dar la máxima difusión a la obra citada a través del Repositorio institucional de la Universidad, el autor **CEDE** a la Universidad Pontificia Comillas, de forma gratuita y no exclusiva, por el máximo plazo legal y con ámbito universal, los derechos de digitalización, de archivo, de reproducción, de distribución y de comunicación pública, incluido el derecho de puesta a disposición electrónica, tal y como se describen en la Ley de Propiedad Intelectual. El derecho de transformación se cede a los únicos efectos de lo dispuesto en la letra a) del apartado siguiente.

### ***3º. Condiciones de la cesión y acceso***

Sin perjuicio de la titularidad de la obra, que sigue correspondiendo a su autor, la cesión de derechos contemplada en esta licencia habilita para:

- a) Transformarla con el fin de adaptarla a cualquier tecnología que permita incorporarla a internet y hacerla accesible; incorporar metadatos para realizar el registro de la obra e incorporar “marcas de agua” o cualquier otro sistema de seguridad o de protección.
- b) Reproducirla en un soporte digital para su incorporación a una base de datos electrónica, incluyendo el derecho de reproducir y almacenar la obra en servidores, a los efectos de garantizar su seguridad, conservación y preservar el formato.
- c) Comunicarla, por defecto, a través de un archivo institucional abierto, accesible de modo libre y gratuito a través de internet.
- d) Cualquier otra forma de acceso (restringido, embargado, cerrado) deberá solicitarse expresamente y obedecer a causas justificadas.
- e) Asignar por defecto a estos trabajos una licencia Creative Commons.
- f) Asignar por defecto a estos trabajos un HANDLE (URL *persistente*).

### ***4º. Derechos del autor.***

El autor, en tanto que titular de una obra tiene derecho a:

- a) Que la Universidad identifique claramente su nombre como autor de la misma
- b) Comunicar y dar publicidad a la obra en la versión que ceda y en otras posteriores a través de cualquier medio.
- c) Solicitar la retirada de la obra del repositorio por causa justificada.
- d) Recibir notificación fehaciente de cualquier reclamación que puedan formular terceras personas en relación con la obra y, en particular, de reclamaciones relativas a los derechos de propiedad intelectual sobre ella.

### ***5º. Deberes del autor.***

- El autor se compromete a:
  - a) Garantizar que el compromiso que adquiere mediante el presente escrito no infringe ningún derecho de terceros, ya sean de propiedad industrial, intelectual o cualquier otro.
  - b) Garantizar que el contenido de las obras no atenta contra los derechos al honor, a la intimidad y a la imagen de terceros.
  - c) Asumir toda reclamación o responsabilidad, incluyendo las indemnizaciones por daños, que pudieran ejercitarse contra la Universidad por terceros que vieran infringidos sus derechos e intereses a causa de la cesión.
  - d) Asumir la responsabilidad en el caso de que las instituciones fueran condenadas por infracción

de derechos derivada de las obras objeto de la cesión.

**6º. Fines y funcionamiento del Repositorio Institucional.**

La obra se pondrá a disposición de los usuarios para que hagan de ella un uso justo y respetuoso con los derechos del autor, según lo permitido por la legislación aplicable, y con fines de estudio, investigación, o cualquier otro fin lícito. Con dicha finalidad, la Universidad asume los siguientes deberes y se reserva las siguientes facultades:

- La Universidad informará a los usuarios del archivo sobre los usos permitidos, y no garantiza ni asume responsabilidad alguna por otras formas en que los usuarios hagan un uso posterior de las obras no conforme con la legislación vigente. El uso posterior, más allá de la copia privada, requerirá que se cite la fuente y se reconozca la autoría, que no se obtenga beneficio comercial, y que no se realicen obras derivadas.
- La Universidad no revisará el contenido de las obras, que en todo caso permanecerá bajo la responsabilidad exclusiva del autor y no estará obligada a ejercitar acciones legales en nombre del autor en el supuesto de infracciones a derechos de propiedad intelectual derivados del depósito y archivo de las obras. El autor renuncia a cualquier reclamación frente a la Universidad por las formas no ajustadas a la legislación vigente en que los usuarios hagan uso de las obras.
- La Universidad adoptará las medidas necesarias para la preservación de la obra en un futuro.
- La Universidad se reserva la facultad de retirar la obra, previa notificación al autor, en supuestos suficientemente justificados, o en caso de reclamaciones de terceros.

Madrid, a 28 de Junio de 2017,

**ACEPTA**

Fdo Elena Fernández García

Motivos para solicitar el acceso restringido, cerrado o embargado del trabajo en el Repositorio Institucional:



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA (ICAI)  
MÁSTER EN INGENIERÍA DE TELECOMUNICACIÓN

# **DISEÑO Y DESARROLLO DE UNA PLATAFORMA DE SOPORTE INTELIGENTE VIRTUAL PARA APRENDIZAJE DE INGLÉS**

Autor: Elena Fernández García  
Director: Miguel Ángel Sanz Bobi

**Madrid**  
Junio 2017



# Agradecimientos

*A mi padre, sin tu creatividad e ingenio esta idea nunca hubiera salido adelante.*

*A mi madre, porque has sido la mejor experta que nadie podría desear. Muchísimas gracias por tu tiempo y dedicación.*

*A Miguel Ángel, director de esta tesis de master, por darme la oportunidad de llevar a cabo este proyecto y por haberme guiado a lo largo de él.*

*A David, coordinador de Trabajos de Fin de Master, por tu apoyo y ayuda durante estos años.*

*A Irene, por haber sabido darle un poco más de vida a este proyecto y por tu actitud positiva en todo momento.*

*A mis compañeros, por vuestra ayuda en esos momentos críticos y por haber podido trabajar día tras día con vosotros para sacar esta carrera adelante.*

*Y a Pierre-Yves, por tu apoyo incondicional a lo largo de estos años.*

*Gracias a todos, porque sin vosotros nunca hubiera podido llegar hasta aquí.*



# DISEÑO Y DESARROLLO DE UNA PLATAFORMA DE SOPORTE INTELIGENTE VIRTUAL PARA APRENDIZAJE DE INGLÉS

**Autor:** Fernández García, Elena.

**Director:** Sanz Bobi, Miguel Ángel.

## RESUMEN DEL PROYECTO

La inteligencia empresarial es un campo que está en crecimiento en este momento. Se ha explotado en muchas disciplinas, pero en educación todavía quedan muchos avances por hacer. Este proyecto ha servido para desarrollar una plataforma de aprendizaje del inglés, que sirve de soporte a alumnos y profesores tanto dentro como fuera del aula. El resultado es una aplicación web capaz de generar ejercicios adaptados a las carencias del alumno y de presentar su evolución.

**Palabras clave:** Inteligencia Empresarial, Educación, Inglés

### 1. Introducción

Los sistemas inteligentes basados en análisis de datos son herramientas que se están adaptando a todos los ámbitos de nuestra sociedad. En los últimos años, hay algunas aplicaciones y sistemas que se han enfocado al ámbito de la educación.

Los sistemas llamados *Intelligent Tutor Systems* que se basan en identificar los pasos mentales que el alumno va dando a la hora de aprender [1]. Estos tutores inteligentes están diseñados para apoyar un aprendizaje eficiente y adaptativo para cada alumno y para ayudar a los profesores a entender en qué momento los alumnos pueden necesitar ayuda específica [2].

Los sistemas inteligentes están encontrando su hueco en el mundo de la educación y todavía queda espacio para desarrollar muchos avances que revolucionen la manera en la que se aprende. Los tutores virtuales están teniendo un momento muy importante dado que están sabiendo adaptarse a las necesidades existentes entre alumnos y profesores.

### 2. Definición del proyecto

Esta tesis de máster ha servido para desarrollar un sistema de apoyo al aprendizaje, tanto para alumnos como para profesores. Es un proyecto que se ha propuesto basándose en una de las problemáticas que se dan en el ambiente educativo actual. Este proyecto se enfoca a las clases que cuentan con alumnos de diferentes edades y niveles, a los que es difícil dar una educación adaptada, o para profesores que deban impartir varias asignaturas y que no sean expertos en cada una de ellas.

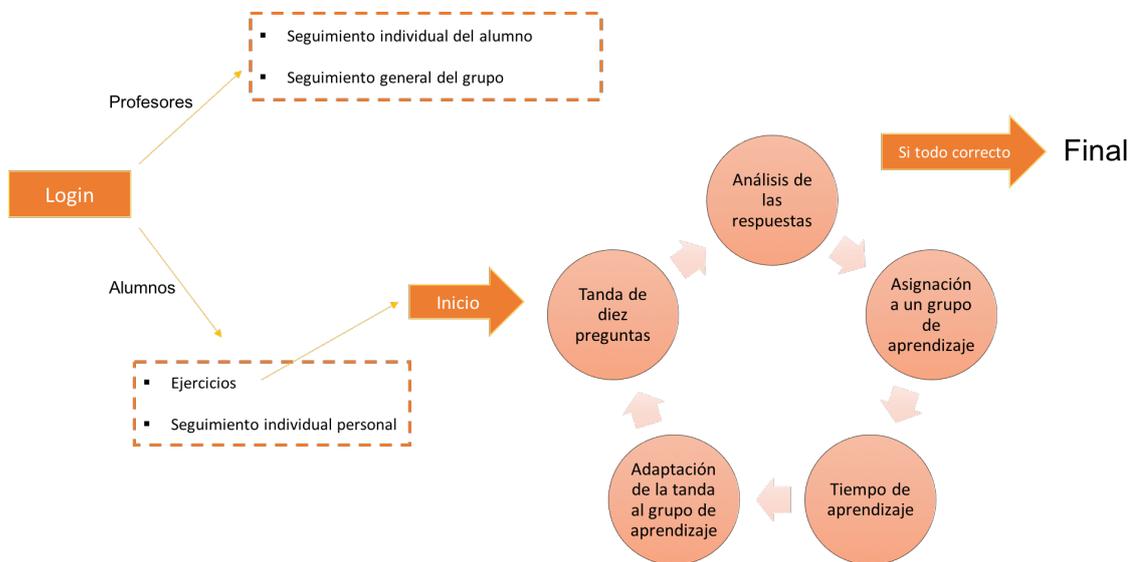
El objetivo principal es el de ofrecer un soporte a profesores y alumnos para el aprendizaje del inglés, en concreto los verbos modales. En ningún momento se trata de sustituir la figura del profesor, si no de complementarla. Los objetivos secundarios son los siguientes:

- Adquisición y generación de conocimiento: En base a entrevistas con un experto y a la información proporcionada por el mismo.
- Definición de perfiles de alumnos: Clasificación de los alumnos en función del conocimiento adquirido previamente.

- Desarrollo de una plataforma web: Como interfaz en la cual los alumnos puedan realizar los ejercicios.
- Seguimiento de la evolución de los alumnos: Para poder observar tanto la tendencia general, cómo la tendencia de cada alumno en particular.

### 3. Descripción del sistema

El sistema que se ha desarrollado se basa en una plataforma web que genera ejercicios de forma aleatoria dependiendo del aprendizaje de cada alumno. En función de los fallos que el alumno tenga, se le clasificará en un grupo de aprendizaje y se propondrá la revisión de los conceptos que no ha entendido. A este tiempo en el que el alumno revisa los conceptos se le ha llamado tiempo de aprendizaje. Tras este tiempo, el alumno podrá realizar una nueva tanda de preguntas que se centrará en aquellos conceptos que no ha entendido.



*Ilustración 1: Proceso del sistema desarrollado*

Cada alumno podrá acceder a su evolución personal, comparada a la media de su clase. También podrá ver cuáles han sido los conceptos que ha fallado en cada uno de los intentos que ha realizado. El profesor, a través de su interfaz, también podrá acceder a estos detalles y ver la evolución de los alumnos que pertenecen a su clase.

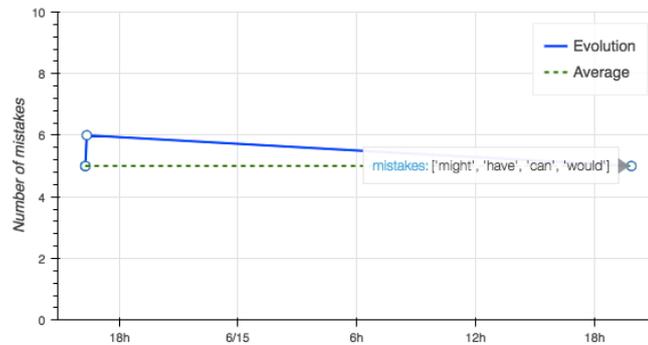
### 4. Resultados

El resultado final de esta tesis de máster es la creación de la plataforma web, llamada Online Tutor, capaz de analizar el aprendizaje de los alumnos y adaptarse a las respuestas que estos dan. Además de la capacidad de adaptarse al aprendizaje del alumno, es importante destacar la capacidad de observar la evolución del alumno. En función de su evolución, el alumno recibirá mensajes motivacionales y se mostrarán cuáles han sido los fallos en cada uno de los intentos realizados.

ONLINE TUTOR Irene Logout

**Welcome to your personal page Irene**  
See your mistake evolution

**You are doing great!**  
Keep working like this



*Ilustración 2: Muestra del resultado final de la plataforma web*

## 5. Conclusiones

Este sistema, diseñado para el aprendizaje del inglés, tiene un modelo muy flexible capaz de adaptarse a otras materias. Simplemente variando la base de datos y la base de conocimiento se podría conseguir que esta plataforma fuera un soporte para toda asignatura, desde matemáticas hasta biología.

Este podría ser el primer paso de una solución real para actualizar el sistema educativo, integrando la tecnología y los métodos que se utilizan actualmente en las aulas.

## 6. Referencias

- [1] W. Ma y O. O. Adesope, «Intelligent Tutoring Systems and Learning Outcomes: A Meta-Analysis,» *Journal of Educational Psychology*, vol. 106, nº 4, pp. 901-918, 2014.
- [2] L. Kurup, A. Joshi y N. Shekhokar, «Intelligent Tutoring System for learning English Punctuation,» de 2016 International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2016.

# DESIGN AND DEVELOPMENT OF AN INTELLIGENT VIRTUAL PLATFORM FOR LEARNING ENGLISH SUPPORT

**Author: Fernández García, Elena.**

Supervisor: Sanz Bobi, Miguel Ángel.

## ABSTRACT

Business Intelligence is a growing trend at this moment. Many fields have been touched by its development. However, there is still a lot of things to do regarding education. This project develops an English learning platform, which gives support to both students and teachers. The result is reflected in a web application able to generate exercises adapted to the concepts the student doesn't understand, and present its evolution.

**Keywords:** Business Intelligence, Education, English

## 1. Introduction

Intelligent systems based on data analytics are tools able to adapt themselves to any field of society. In the last years, some of these tools have focused on education.

Intelligent Tutor Systems identify mental steps that students go through when they are learning [1]. These tutoring systems are designed to develop an efficient and adaptive learning system for each student and to help teachers understand when their students need specific help [2].

Business Intelligence is being able to find its place in educational systems and there is still place to develop many more systems that might change the way we learn. Virtual tutoring systems are having a momentum as they can find the existing needs between teachers and students.

## 2. Project definition

This master thesis develops a learning support system focused on both students and teachers. It is a project based on identified issues such as the fact that teachers have many students and it is difficult for them to offer personalized help to all of them. It is also developed for those classrooms with students from different ages and levels, because it is difficult to adapt the classes for them, and for teachers that must teach many subjects and are not experts in all of them.

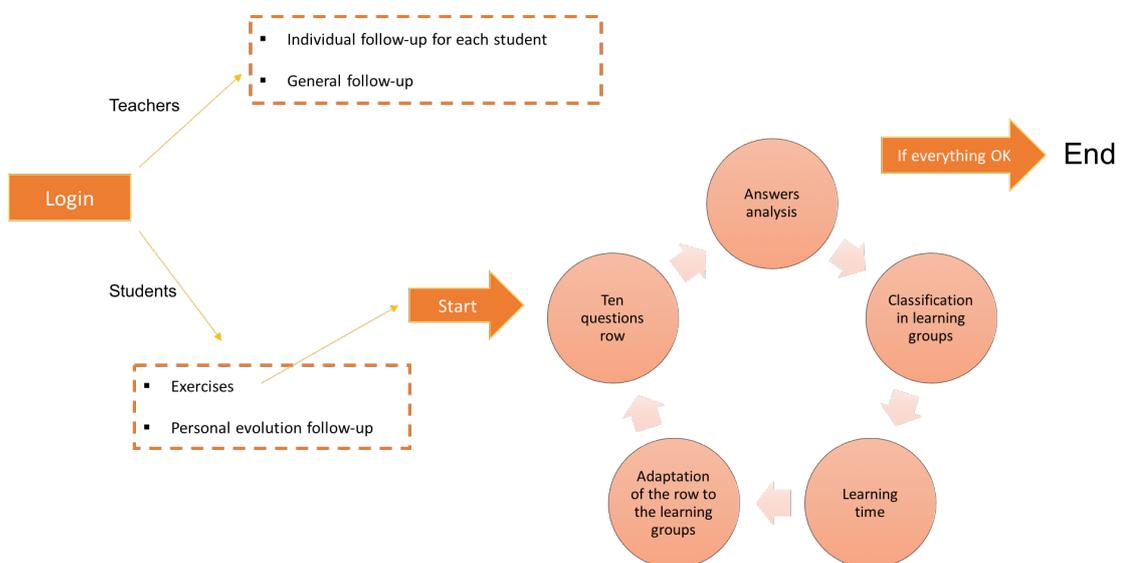
The main objective is to offer support to learn English, specifically the concept of modal verbs. This system doesn't intend to be a substitute of teachers, but to complement them. Secondary objectives are the following:

- Knowledge acquisition and generation: Based on interviews with an expert.
- Profile definition: Classification of students depending on previous knowledge acquisition.
- Web platform development: As an interface where students are able to do their exercises.

- Evolution of each student: To have information about global and individual trends.

### 3. System description

The developed system is based on a web platform that generates random exercises depending on the learning curve of each student. Depending on the mistakes, the student will be classified in different groups and will be proposed the review of the concepts he didn't understand. The time devoted to review the concepts is called learning time. After that time, the student will be able to do more exercises focusing on the concepts that weren't understood.



*Ilustración 3: Process of the developed system*

Each student will be able to access its individual evolution, compared to the class average. He will also see which are the concepts he didn't understand every time he tried to do the exercises. The teacher, through his interface, will also be able to access those details and see the evolution of the students that belong to his class.

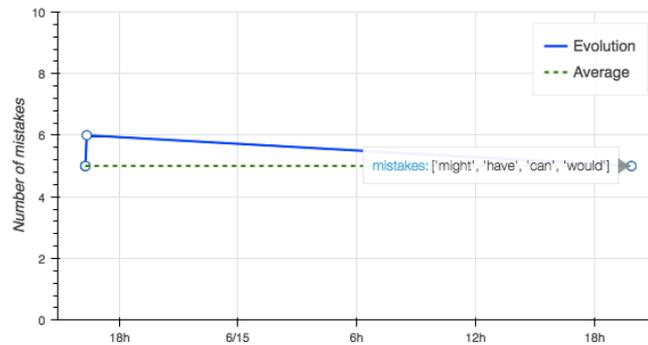
### 4. Results

The result of this master thesis is the development of a web platform, called Online Tutor, able to analyze the learning process of each student and adapt itself to the answers they give. In addition, it is important to emphasize the possibility of seeing the evolution of the student. Depending on its evolution, students will receive motivational messages and will be shown which their mistakes were in each trial.

ONLINE TUTOR Irene Logout

**Welcome to your personal page Irene**  
See your mistake evolution

**You are doing great!**  
Keep working like this



*Ilustración 4: View of the evolution of a student*

## 5. Conclusions

The system, designed to learn English, has a very flexible model and it would be easy to adapt it to other subjects. By changing the data base and the knowledge base, it could give support to any other subject, from Mathematics to Biology.

This could be the first step to develop a real solution that could update the educational system by integrating technology and the methodology that is now applied in the classrooms.

## 6. References

- [1] W. Ma y O. O. Adesope, «Intelligent Tutoring Systems and Learning Outcomes: A Meta-Analysis,» *Journal of Educational Psychology*, vol. 106, nº 4, pp. 901-918, 2014.
- [2] L. Kurup, A. Joshi y N. Shekhokar, «Intelligent Tutoring System for learning English Punctuation,» de 2016 International Conference on Computing Communication Control and Automation (ICCCBEA), Pune, India, 2016.

## Índice de la memoria

<b>Capítulo 1. Introducción.....</b>	<b>6</b>
1.1 Motivación y contextualización del proyecto .....	6
1.2 Business Intelligence y su utilidad en el contexto actual.....	6
<b>Capítulo 2. Descripción de las Tecnologías.....</b>	<b>10</b>
2.1 Microsoft Excel.....	10
2.2 Matlab .....	10
2.3 Weka .....	11
2.4 Python .....	12
2.4.1 Django.....	12
2.4.2 Pycharm .....	13
2.5 Bootstrap.....	14
2.6 Google Chrome .....	14
<b>Capítulo 3. Estado de la Cuestión .....</b>	<b>15</b>
3.1 Estado del arte.....	15
3.1.1 Business Intelligence en entornos educativos .....	15
3.1.2 Business Intelligence para el aprendizaje del inglés .....	17
3.2 Conclusiones extraídas del estado del arte.....	18
<b>Capítulo 4. Definición del Trabajo.....</b>	<b>20</b>
4.1 Justificación .....	20
4.2 Objetivos .....	21
4.3 Metodología .....	21
4.4 Planificación y Estimación Económica.....	22
<b>Capítulo 5. Sistema Desarrollado.....</b>	<b>23</b>
5.1 Diseño y estrategia del sistema .....	24
5.1.1 Elementos del sistema .....	27
5.2 Adquisición y generación del conocimiento del sistema.....	31
5.2.1 Definición del perfil de los alumnos .....	33
5.2.2 Clasificación en grupos de aprendizaje.....	35
5.3 Definición de la base de conocimientos.....	37

5.3.1 Reglas de inferencia.....	42
5.4 Implementación.....	43
5.4.1 Generación aleatoria de frases.....	44
5.4.2 Generación de ejercicios.....	49
5.4.3 Clasificación en grupos de aprendizaje.....	51
5.4.4 Análisis de la evolución de los alumnos .....	56
5.4.5 Vista de profesor .....	58
5.5 Diseño gráfico de la aplicación.....	59
<b>Capítulo 6. Comprobación del sistema.....</b>	<b>62</b>
6.1 Casos ejemplo .....	62
6.1.1 Caso de un alumno.....	63
6.1.2 Caso de un profesor .....	66
6.1.3 Caso de un nuevo desarrollador.....	68
6.2 Dificultades encontradas .....	68
<b>Capítulo 7. Conclusiones y Trabajos Futuros .....</b>	<b>70</b>
<b>Capítulo 8. Bibliografía .....</b>	<b>72</b>
<b>Anexo A – Códigos de respuesta .....</b>	<b>75</b>
<b>ANEXO B – Grupos de Aprendizaje .....</b>	<b>77</b>
<b>ANEXO C – Grupos de Aprendizaje y Recomendaciones.....</b>	<b>78</b>

## *Índice de ilustraciones*

Ilustración 1: Proceso del sistema desarrollado .....	11
Ilustración 2: Muestra del resultado final de la plataforma web.....	12
Ilustración 3: Process of the developed system .....	14
Ilustración 4: View of the evolution of a student.....	15
Ilustración 5: Arquitectura del sistema [19].....	18
Ilustración 6: Proceso base del sistema.....	25
Ilustración 7: Proceso del sistema final .....	27
Ilustración 8: Elementos previos al funcionamiento del sistema.....	27
Ilustración 9: Elementos previos al funcionamiento del sistema y herramientas utilizadas	29
Ilustración 10: Componentes de interacción con el sistema .....	29
Ilustración 11: Componentes de interacción con el sistema y herramientas utilizadas .....	31
Ilustración 12: Perfil de los alumnos .....	34
Ilustración 13: Perfil de los alumnos con un grado de similitud del 98% .....	35
Ilustración 14: Interfaz de WEKA tras añadir los datos a la plataforma .....	40
Ilustración 15: Extracto del árbol de decisión resultante .....	41
Ilustración 16: Proceso de generación aleatoria de frases .....	45
Ilustración 17: Ejemplo de creación de un proyecto con Django .....	46
Ilustración 18: Creación de una aplicación dentro de un proyecto de Django .....	47
Ilustración 19: Extracto del archivo Models.py de la aplicación de frases.....	48
Ilustración 20: Interfaz de administrador de Django .....	48
Ilustración 21: Vista de un ejercicio de la aplicación .....	50
Ilustración 22: Vista de la aplicación al acabar un ejercicio.....	51
Ilustración 23: Extracto del código del algoritmo de clasificación.....	52
Ilustración 24: Vista de la aplicación en la opción See the answers.....	53
Ilustración 25: Vista de revisión de conceptos de la aplicación .....	54
Ilustración 26: Extracto del código de generación de frases en función del Grupo de Aprendizaje.....	55
Ilustración 27: Ejemplo de evolución de un alumno .....	56

*ÍNDICE DE ILUSTRACIONES*

---

Ilustración 28: Evolución del alumno y muestra de los fallos cometidos.....	57
Ilustración 29: Índice de la aplicación para un profesor .....	58
Ilustración 30: Evolución de un alumno y comparación con la media de la clase .....	59
Ilustración 31: Vista de índice de la aplicación desde un dispositivo pequeño .....	60
Ilustración 32: Logo de la aplicación.....	61
Ilustración 33: Vista del sistema de autenticación.....	62
Ilustración 34: Vista de índice para un alumno .....	63
Ilustración 35: Vista de las preguntas propuestas a un alumno .....	63
Ilustración 36: Resultados del ejercicio de un alumno .....	64
Ilustración 37: Extracto de las preguntas acertadas y falladas del alumno.....	64
Ilustración 38: Vista de revisión de conceptos .....	65
Ilustración 39: Vista del perfil personal del alumno.....	66
Ilustración 40: Vista de un profesor.....	67
Ilustración 41: Evolución de un alumno, vista de profesor .....	67

## *Índice de tablas*

Tabla 1: Comparación del sistema desarrollado y el estado del arte .....	19
Tabla 2: Ejemplo de análisis de los Grupos de Aprendizaje .....	36
Tabla 3: Extracto de la tabla de datos introducida en WEKA .....	37

## **Capítulo 1. INTRODUCCIÓN**

### ***1.1 MOTIVACIÓN Y CONTEXTUALIZACIÓN DEL PROYECTO***

Esta tesis de máster tiene básicamente una motivación social basada en una de las problemáticas que se dan en el ambiente educativo actual. El trabajo se enfoca a las clases que cuentan con alumnos de diferentes edades y niveles, a los que es difícil dar una educación adaptada, o para profesores que deban impartir varias asignaturas y que no sean expertos en cada una de ellas.

El objetivo principal de esta tesis de máster es de ser un apoyo para el profesor y el alumno tanto dentro como fuera del aula, en ningún momento trata de ser un sustituto de la figura del profesor.

Se trata de una plataforma que ofrece ayuda al alumno mediante la propuesta de ejercicios sobre un tema en particular. En este caso se ha decidido realizar una plataforma de ayuda al aprendizaje de inglés, más concretamente sobre el aprendizaje de los verbos modales.

Igualmente, la plataforma proporcionará realimentación al profesor a través del análisis de las respuestas de los alumnos de su clase. De esta manera el profesor sabrá a quien debe dar qué tipo de atención, si es preferible reforzar conocimientos a través de tutorías o si hay algún concepto que toda la clase no haya entendido y que se considere importante volver a impartir.

### ***1.2 BUSINESS INTELLIGENCE Y SU UTILIDAD EN EL CONTEXTO ACTUAL***

Business Intelligence es un término que surgió en los años 80 y fue utilizado por la consultora Gartner Group por primera vez. Este término definía la capacidad para acceder a la

información de una base de datos y explorarla. De tal manera el usuario puede analizar esa información y desarrollar teorías que serán esenciales para la toma de decisiones en una empresa [1]. Business Intelligence es una manera de afrontar los problemas empresariales estratégicos, utilizando los datos existentes de manera inteligente [2]. Este uso de los datos se puede realizar igualmente para temas que no tengan que ver con empresas, como por ejemplo para tratar problemas sociales. Los usos de la Inteligencia Empresarial son muy amplios, por ello en muchos casos esto ha pasado a llamarse únicamente *Analytics*, englobando así todos los temas que se puedan ver incluidos en la propuesta de soluciones basadas en el análisis de datos.

Las características principales de Business Intelligence son las siguientes [3]:

- Reconocimiento de una experiencia
- Análisis de datos en su contexto
- Capacidad de extraer datos e integrarlos a partir de múltiples fuentes
- El procesamiento de la información obtenida para extraer la información útil
- La búsqueda de relaciones y trabajo de hipótesis

Cada una de estas características es esencial y contribuye a que el análisis de datos tenga sentido en cuando a la adquisición y utilización de conocimiento para la mejora de procesos, sistemas, etc., en cualquier ámbito.

Se considera que las reglas de oro de Business Intelligence son las siguientes [1]:

- Calidad y exactitud del dato
- Consistencia del dato
- Disponibilidad del dato necesario en el momento adecuado
- Coger la información adecuada en el momento adecuado de la fuente adecuada

La calidad de los datos es muy importante y afecta en muchos sentidos al sistema de análisis que se quiera construir. Básicamente, los datos son la clave del éxito de este tipo de sistemas, dado que son la fuente de la que nos fiamos para realizar ciertas hipótesis. Si la fuente no es correcta, las hipótesis y los resultados que se obtengan de ella no serán válidos [4]. Las características más importantes que un dato debe tener para ser considerado como apto para su utilización en un sistema son los siguientes [5]:

- Confidencialidad  
Esta característica se refiere a que sólo las personas autorizadas pueden acceder a los datos. Esto evita interceptaciones y que la información pueda ser leída por una tercera persona.
- Integridad  
Es una cualidad en la que se comprueba que el dato no ha sido alterado, permanece el contenido original.
- Disponibilidad  
Es la capacidad de poder acceder a los datos cuando el usuario lo requiera, evitando pérdidas o bloqueos de información.
- Autenticación

La verificación de que el documento ha sido elaborado o pertenece a quien se indica en el documento.

Para hacernos una idea de la magnitud que tiene este sector, el Ministerio de Energía, Turismo y Agenda Digital señalaba en 2013 que el análisis de datos generaría a nivel mundial 4,4 millones de puestos de trabajo en 2015. Asimismo, afirmaba que un 64% de las organizaciones han invertido o cuentan invertir en Big Data durante el año 2013 [6].

## **Capítulo 2. DESCRIPCIÓN DE LAS TECNOLOGÍAS**

### **2.1 *MICROSOFT EXCEL***

Microsoft Excel es una herramienta que tiene muchas utilidades, en su mayor parte para el análisis de datos. Es una aplicación muy fácil de utilizar y de conseguir, la Universidad Pontificia de Comillas pone a disposición de los alumnos todo el repertorio de aplicaciones de Microsoft Office. Además, cuenta con la posibilidad de crear pequeños programas con la ayuda de un editor, Macros, que permite la utilización del lenguaje VBA (Visual Basic Applications).

En el contexto de este proyecto se ha utilizado para crear la base de conocimiento. La base de conocimiento es una base de datos en la que se ha almacenado información sobre las respuestas que han dado los alumnos a cada una de las preguntas que se le han propuesto. Igualmente, se ha utilizado para comprobar la similitud de las respuestas de un perfil de alumnos mediante filtros y de esa manera poder asignar a cada alumno un grupo con consignas específicas y adaptadas.

También se ha utilizado para hacer un prototipo del algoritmo de generación de frases, a través de Visual Basic, antes de implementarlo en un lenguaje de programación más adaptado.

La versión de Excel utilizada ha sido 15.32 (170309).

### **2.2 *MATLAB***

Matlab es una plataforma creada para resolver problemas de ingeniería y ciencia. Matlab está basado en matrices, una forma sencilla para expresar matemáticas computacionales.

Incluye gráficos que facilitan la visualización de los datos y la posibilidad de obtener información a partir de los mismos [7].

En este caso, la universidad pone a disposición de los alumnos esta plataforma, de manera que el acceso es bastante sencillo. A lo largo de la carrera, se nos ha enseñado a utilizar esta plataforma en varias asignaturas por lo que su utilización no ha supuesto un problema. La versión utilizada es R2016B for academic use.

Matlab se ha utilizado durante la fase de análisis, tal y como se explicará más adelante.

## 2.3 WEKA

Weka es una aplicación basada en algoritmos de *machine learning*, usada para realizar funciones de minería de datos. Se trata de una aplicación *Open Source* a la que se le pueden cargar archivos para procesarlos [8]. Los archivos deben estar en formato *.arff* y con una estructura especial para que el algoritmo pueda analizar dicho archivo. A continuación se muestra un ejemplo de la forma que debe tener el fichero [9]:

```
@RELATION iris
@ATTRIBUTE sepallength NUMERIC
@ATTRIBUTE sepalwidth NUMERIC
@ATTRIBUTE petalwidth NUMERIC
@ATTRIBUTE class {Iris-setosa,Iris-versicolor,Iris-virginica}
@DATA
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
```

El apartado *@RELATION* solamente indica el nombre de la relación que se quiere estudiar. *@ATTRIBUTE* indica los distintos atributos que se quieren tener en cuenta para desarrollar

las relaciones, igualmente se debe escribir que tipo de formato debe esperar el programa. En este caso se ha puesto NUMERIC, lo que quiere decir que los valores de los atributos van a ser numéricos. Se debe añadir un atributo con el nombre class o clase, que será la clasificación final que se quiere hacer de los datos. En este ejemplo se estudian los distintos tipos de Iris en función de unos parámetros (los atributos).

Después de @DATA, se exponen los datos separados por comas y con los atributos en el mismo orden en el que se han declarado anteriormente.

WEKA es una herramienta muy útil para obtener reglas de inferencia y árboles de decisión, entre otras funcionalidades. También es bastante útil cuando se quiere verificar la información dada por un experto, ya que permite comprobar el error con el cual se pueden clasificar los datos. Durante el desarrollo del proyecto, la aplicación se ha usado básicamente para esos fines.

## **2.4 PYTHON**

Python es un lenguaje de programación que tiene como objetivo el favorecer la creación de un código legible [10]. En esta tesis de máster se ha utilizado la funcionalidad de programación orientada a objetos. Se trata de un lenguaje de programación de código abierto, lo que facilita la búsqueda de información para su comprensión y para su aprendizaje.

Dado que el objetivo de la tesis ha sido presentar una plataforma web a la que alumnos y profesores puedan acceder, se ha utilizado un framework de Python que ofrece facilidades para crear dichas funcionalidades. Se trata de un framework llamado Django.

### **2.4.1 DJANGO**

Django es un framework de alto nivel escrito en Python que ayuda al diseño y creación de aplicaciones web. Los tres puntos que caracterizan a Django son los siguientes [11]:

- Rapidez: Django está diseñado para que los desarrolladores puedan implementar sus aplicaciones lo más rápido posible.
- Seguridad: La seguridad es una preocupación de Django y ofrece a los desarrolladores ayuda para evitar los errores más comunes a la hora de crear aplicaciones web.
- Escalabilidad: Importante para las aplicaciones que van ganando seguidores o usuarios a lo largo del tiempo, Django ofrece flexibilidad y escalabilidad.

Para poder hacer uso de Django, se ha tenido que contar con la ayuda de varios tutoriales. Dichos tutoriales, además de mostrar el uso de Django, explican cómo programar en Python y han sido muy útiles para el desarrollo de la aplicación. [12], [13].

#### **2.4.2 PYCHARM**

Con el objetivo de poder seguir los tutoriales de la mejor manera posible, se ha decidido utilizar el IDE que estos han utilizado. Pycharm es un IDE diseñado para programar proyectos de Python [14]. Estas son algunas de las características de Pycharm:

- Asistencia inteligente en Python: Pycharm tiene la capacidad de completar las sentencias cuando se está programando. Esto hace que la programación sea rápida y precisa.
- Framework para desarrollo web: Pycharm soporta Django, lo que ha sido muy útil para el desarrollo del proyecto y ha facilitado la interoperabilidad de las herramientas utilizadas.

- Desarrollo cruzado: Aunque Pycharm sea un IDE de Python, también se puede programar en HTML o CSS, entre otros. A la hora de diseñar una aplicación web este aspecto ha sido clave ya que ha facilitado mucho la integración de lenguajes de programación.

## **2.5 BOOTSTRAP**

Bootstrap es una herramienta muy útil para dar un aspecto atractivo a aplicaciones web. Es capaz de hacer que las aplicaciones sean responsivas, es decir, que sean capaces de adaptarse a cualquier dispositivo. Este aspecto es importante dado que, según está evolucionando la sociedad, ya no sólo nos conectamos a internet y usamos aplicaciones a través del ordenador, si no que también podemos hacerlo a través de smartphones o tablets [15].

Bootstrap ofrece la posibilidad de personalizar los templates HTML simplemente añadiendo características a las etiquetas del documento. En esta tesis se ha utilizado Bootstrap a través de CDN, es decir, que a través de unos scripts que se han añadido en la cabecera de los documentos HTML se ha podido acceder a todas las funcionalidades de la herramienta.

## **2.6 GOOGLE CHROME**

Es necesario mencionar Google Chrome dado que es el navegador que se ha utilizado para la visualización de la aplicación web mientras ha sido desarrollada. Se trata de un navegador diseñado por Google y adaptable a cualquier tipo de sistema operativo. En este caso se ha utilizado Google Chrome para MacOS.

## **Capítulo 3. ESTADO DE LA CUESTIÓN**

### **3.1 ESTADO DEL ARTE**

La inteligencia empresarial ha resultado ser una de las herramientas más valoradas a la hora de tomar decisiones en el entorno de los negocios. Su evolución ha sido notable en los últimos años y sigue siendo una tendencia al alza. Se trata de una herramienta de mejora para las organizaciones, dado que un mejor conocimiento del negocio a todos los niveles lleva a una toma de decisiones más eficiente [1].

Los pilares de la inteligencia empresarial se han encontrado en el campo de DSS (Sistemas de Soporte a la Decisión), es la disciplina de Sistemas de Información que se basa en la mejora de la toma de decisiones desde un punto de vista gerencial. La inteligencia empresarial combina el tratamiento de datos estructurados y sin estructurar para su conversión en información útil. De esta manera los gerentes pueden estar mejor informados y, por consiguiente, tomar mejores decisiones [16].

En cualquier caso, el uso de inteligencia empresarial está ligado a un entorno de negocios. Sin embargo, se trata de una disciplina que se puede utilizar en muchos otros campos, como es el caso de la medicina o la educación.

#### **3.1.1 BUSINESS INTELLIGENCE EN ENTORNOS EDUCATIVOS**

Business Intelligence es un importante avance en el mundo de la tecnología, y cada vez afecta a más áreas. Aunque nuestro mundo cada vez está más basado en sistemas inteligentes, muy pocas de estas aplicaciones se han desarrollado mirando al mundo educativo. Hace ya algunos años se proponían algunos de los retos más importantes que los sistemas inteligentes podrían resolver [17]:

- Tutores virtuales para cada alumno
- Evaluación de las competencias del siglo XXI: Siendo estas auto-evaluación y trabajo en equipo entre otras.
- Análisis de la interacción: Analizar el aprendizaje individual, teniendo en cuenta el contexto social y educativo, así como los intereses personales del alumno.
- Creación de oportunidades para clases globales: a través de la conectividad y accesibilidad global.

Hay unos sistemas llamados *Intelligent Tutor Systems* que se basan en identificar los pasos mentales que el alumno va dando a la hora de aprender [18]. Los tutores inteligentes están diseñados para apoyar un aprendizaje eficiente y adaptativo para cada alumno y ayudar a los profesores a entender en qué momento los alumnos pueden necesitar ayuda específica [19]. Las características de cualquier tutor inteligente son las siguientes:

- Evaluación de las acciones del alumno
- Modelos adaptativos para proponer posibles soluciones al ejercicio
- Propone realimentación o indicios para completar la actividad

Otras aplicaciones de Business Intelligence en educación, son los sistemas llamados de aprendizaje profundo. Estos sistemas se basan en emular el comportamiento humano. Herramientas como Watson de IBM, son soluciones cognitivas basadas en el aprendizaje, preferencias y aptitudes de cada alumno. Existen estudios que demuestran que una atención

personalizada y adaptada a cada alumno permite que estos puedan crecer y conseguir los logros que se hayan propuesto de manera más sencilla [20].

También se están desarrollando redes sociales con el objeto de que los alumnos colaboren entre ellos para rellenar las lagunas que unos u otros puedan tener [21]. Las aplicaciones que se pueden crear para la educación con base en Business Intelligence son muy variadas y en este caso la imaginación es el límite para crear herramientas útiles que sean capaces de ayudar a la sociedad y hacer que las personas colaboren para poder aprender.

En conclusión, los sistemas inteligentes están encontrando su hueco en el mundo de la educación y todavía queda espacio para desarrollar muchos avances que revolucionen la manera en la que se aprende. Los tutores virtuales están teniendo un momento muy importante dado que están sabiendo adaptarse a las necesidades existentes entre alumnos y profesores.

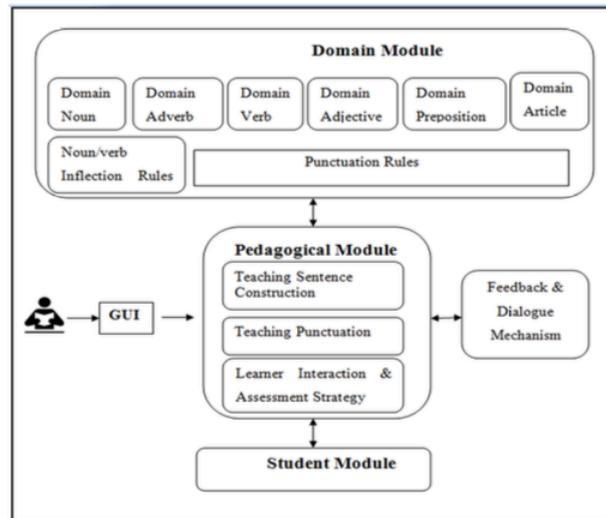
### **3.1.2 BUSINESS INTELLIGENCE PARA EL APRENDIZAJE DEL INGLÉS**

Dado que el inglés es un idioma que se estudia a nivel mundial, la problemática del aprendizaje del inglés ya se ha tratado en algunos trabajos. Pero como es un lenguaje muy extenso, estos tutores inteligentes sólo se han ocupado de una parte del idioma.

Por ejemplo, en [19] se trata el desarrollo de un sistema de tutoría inteligente para el aprendizaje de los métodos de puntuación en inglés. Este sistema en concreto tiene la estructura que se muestra en la Ilustración 5. Los bloques se dividen en:

- Dominio: Información sobre el módulo de aprendizaje que se va a tratar
- Módulo de pedagogía

- Módulo del estudiante



*Ilustración 5: Arquitectura del sistema [19]*

Además de los módulos mencionados anteriormente, también incluye la generación de frases, una fase de retroalimentación al alumno y una interfaz con el estudiante.

También hay trabajos que se han basado en minería de datos para poder proponer un sistema de aprendizaje adaptativo a la hora de aprender inglés. En [22] se muestra cómo a partir de la minería de datos se puede obtener información que no se puede percibir a simple vista y reglas, para la construcción de un tutor de inglés. Además, el hecho de poder analizar datos reales hace que el impacto en investigación en el ámbito de la educación sea más eficaz.

### **3.2 CONCLUSIONES EXTRAÍDAS DEL ESTADO DEL ARTE**

Los casos que se han ido mostrando a lo largo de este capítulo son sólo algunos de los ejemplos que muestran lo que los sistemas inteligentes pueden aportar a la educación, y más concretamente al estudio del inglés. Son estudios publicados en el último año, lo que es una muestra de la novedad de este tipo de aplicaciones. Tal y como se ha comentado

anteriormente, estos avances son importantes y dan pie a la creación de avances aún más importantes en el campo de la educación.

Finalmente, se van a comparar las características de los sistemas comentados en el estado del arte a las características del tutor inteligente desarrollado en esta tesis de máster.

*Tabla 1: Comparación del sistema desarrollado y el estado del arte*

	<b>Estado del arte</b>	<b>Sistema desarrollado</b>
<i>Evaluación de las acciones</i>	√	√
<i>Propuesta de soluciones adaptadas</i>	√	√
<i>Realimentación</i>	√	√
<i>Minería de datos</i>	√	√
<i>Generación de frases</i>	√	√
<i>Visualización de la evolución</i>	-	√
<i>Motivación al alumno</i>	-	√
<i>Interfaz para profesor</i>	-	√
<i>Adaptabilidad del modelo a otras materias</i>	-	√

Esta comparación es orientativa dado que los trabajos estudiados no dan información exhaustiva sobre las plataformas desarrolladas y no se ha tenido acceso a dichos sistemas. El símbolo ‘-’ significa que no se tiene información de que esa funcionalidad haya sido implementada.

## **Capítulo 4. DEFINICIÓN DEL TRABAJO**

### **4.1 JUSTIFICACIÓN**

Esta tesis de máster se ha desarrollado para ofrecer un soporte tanto a profesores como alumnos en las aulas y fuera de ellas. El diseño del sistema surgió por una preocupación, aquella de ayudar a los colegios en los cuales los profesores deben dar varias materias y no son expertos en todas ellas. También se diseñó pensando en las aulas que cuentan con alumnos de distintos niveles.

Puede parecer que estos casos sean muy lejanos, pero no es así. Tan sólo hay que irse a algunos pueblos españoles para constatar que esto es un problema. Esta aplicación pretende ser un soporte al aprendizaje de los alumnos, proponiendo retos adaptados a su nivel y a su curva de aprendizaje.

En el desarrollo de esta tesis de máster, se ha tomado como punto de partida el aprendizaje del inglés porque se ha tratado de la materia en la que se podría contar con la ayuda de un experto de manera sencilla. Pero este sistema se podría extrapolar a cualquier materia. Simplemente variando la base de datos y la base de conocimiento se podría conseguir que esta plataforma fuera un soporte para toda asignatura, desde matemáticas hasta biología.

La primera piedra para construir este proyecto educativo ya ha sido puesta. Queda seguir adelante para conseguir que esta aplicación sea una solución real y pueda ayudar a todos aquellos profesores que tienen dificultades a la hora de dar atención personalizada a sus alumnos en ciertos temas, y a los alumnos que desean continuar su aprendizaje resolviendo nuevos retos.

## **4.2 OBJETIVOS**

Los objetivos del sistema se han ido comentando a lo largo de esta introducción. El objetivo general perseguido en este trabajo es desarrollar un sistema de ayuda tutorial, denominado “Online Tutor”, que ayude a los alumnos en el aprendizaje del idioma inglés y a sus profesores a ver su progreso y sus dificultades en el aprendizaje. En la versión desarrollada sólo se ha tratado el tema del uso de verbos modales en inglés. Sin embargo, el diseño realizado es lo suficientemente flexible como para contemplar otros aspectos de la enseñanza del inglés.

Como objetivos secundarios se pueden enunciar los siguientes:

- Adquisición y generación de conocimiento: En base a entrevistas con un experto y a información proporcionada por el mismo.
- Definición de perfiles de alumnos: Clasificación de los alumnos en función del conocimiento adquirido previamente.
- Desarrollo de una plataforma web: En la cual los alumnos puedan realizar los ejercicios.
- Seguimiento de la evolución de los alumnos: Para poder observar tanto la tendencia general, cómo la tendencia de cada alumno en particular.

## **4.3 METODOLOGÍA**

Para poder conseguir los objetivos mencionados en el apartado anterior, se ha seguido una metodología de entrevistas con expertos. Estas entrevistas han sido claves para entender que áreas necesitaban atención más rápidamente y que la solución aportada por el sistema fuera de real utilidad.

Tras entrevistas con expertos, se ha diseñado el sistema, tomando el tiempo necesario para entender cuál iba a ser su función principal y poder llevarla a cabo. Tras el diseño, se ha implementado el sistema siempre teniendo en cuenta la funcionalidad del sistema.

El diseño y la implementación han ido de la mano constantemente, dado que al realizar cada uno de los hitos marcados a lo largo del proyecto no se ha dudado en diseñar nuevas funcionalidades que hagan de la aplicación una opción real para el soporte del aprendizaje.

#### ***4.4 PLANIFICACIÓN Y ESTIMACIÓN ECONÓMICA***

Este sistema se ha construido basándose en herramientas y tecnologías de código abierto y accesibles a cualquier persona. Esto hace que el coste sea bastante reducido dado que lo único que se podría contabilizar son las horas invertidas.

Tomando cómo media 15 horas de trabajo por semana a lo largo de las, aproximadamente, 40 semanas de trabajo, se puede estimar que este sistema ha necesitado de 600 horas de trabajo. Para una única persona que se ocupe del proyecto, teniendo en cuenta que su nivel de estudios es de master, se podrían contar 8€/hora. Esto significa que el coste del proyecto tal y cómo se presenta en este documento sería de **4.800€**.

## **Capítulo 5. SISTEMA DESARROLLADO**

Para desarrollar esta tesis de máster se ha contado con la ayuda de un experto, sin este apoyo el sistema no podría haberse desarrollado ya que el experto es la persona que ha proporcionado una gran parte del conocimiento del sistema. Se barajaron varias temáticas para el sistema, pero la decisión de enfocarlo hacia el aprendizaje del inglés se basó en la ayuda que se podría conseguir para comprender el proceso de aprendizaje de un alumno.

Para este sistema se ha contado con la ayuda de una profesora de inglés de nivel Bachillerato. Tras tener varias entrevistas con ella, se estableció que la parte que ella identificaba como más conflictiva para los alumnos era el aprendizaje y uso de los modales en la lengua inglesa. Por esta razón se decidió tomar esta parte como punto de partida del sistema a desarrollar, ya que es la parte en la que el proyecto podría tener más impacto para los alumnos que lo utilicen.

Este capítulo se va a dividir en varias secciones en las que se va a ir relatando cuál ha sido el procedimiento para desarrollar el sistema de aprendizaje.

Primero se tratará el diseño del sistema y la estrategia del mismo, en esta parte se relatarán los aspectos claves del sistema y es importante para la comprensión de su funcionamiento.

Se continuará con la explicación de la adquisición y la generación del conocimiento necesario para realizar la aplicación. El procedimiento para clasificar a los alumnos, así como la definición del objetivo de esta clasificación se tratarán en este apartado.

La definición de la base de conocimiento será tratada en el siguiente apartado, haciendo hincapié en las reglas de inferencia obtenidas. Estas reglas de inferencia son la clave que definen el funcionamiento del algoritmo y que determinan su comportamiento en cada caso.

Seguidamente se explicará todo el proceso de implementación y cada una de las funcionalidades que se han diseñado para la aplicación final, tanto desde el punto de vista de un alumno como de un profesor.

Este capítulo finalizará por la descripción del diseño gráfico del sistema.

## **5.1 DISEÑO Y ESTRATEGIA DEL SISTEMA**

Para diseñar el sistema, se han tenido que determinar algunos aspectos clave de su funcionamiento. Por ejemplo, se ha de tener muy claro que la plataforma de aprendizaje **no pretende sustituir a la figura del profesor**, si no que se debe tratar como un apoyo tanto para el alumno como para el profesor. El conocimiento que un profesor puede transmitir a sus alumnos no se puede comparar con una plataforma de aprendizaje.

De este modo, para diseñar el sistema se va a tener muy en cuenta la figura del profesor. El sistema ha ido evolucionando según se han ido alcanzando los hitos definidos. El sistema original fue diseñado pensando en un bucle. El alumno accede a la plataforma y se le proponen diez ejercicios sobre los modales. En función de las respuestas que el alumno dé, a continuación se le proponen otros diez ejercicios teniendo en cuenta los puntos flacos, es decir, los conceptos que el alumno no ha entendido bien. De esta manera el alumno puede practicar los ejercicios que resultan más complicados. El sistema se adapta a cada alumno y a su evolución.



*Ilustración 6: Proceso base del sistema*

En la Ilustración 6 se puede observar el bucle base. En él se accede a la plataforma y se plantean las diez preguntas de forma **aleatoria**. Tras un análisis de las respuestas que se han dado, se asigna al alumno a lo que se ha llamado *Grupo de Aprendizaje*. Los Grupos de Aprendizaje son clusters a los cuales se asignan los alumnos en función de sus respuestas. Más adelante se explicará cuál ha sido la lógica a la hora de crear dichos Grupos de Aprendizaje.

El hecho de ser asignado a un grupo de aprendizaje supone que la siguiente tanda va a estar condicionada. Se van a proponer ejercicios sobre los conceptos que el alumno no ha sabido contestar correctamente. Pero antes de pasar a la parte de los ejercicios, es necesario que el alumno disponga de un tiempo determinado para poder asimilar los conceptos que previamente no ha entendido. Por ello, se le da la opción al alumno de poder aprender, y durante este tiempo se le ofrecerán unas fichas definiendo y explicando los casos de uso del modal en cuestión.

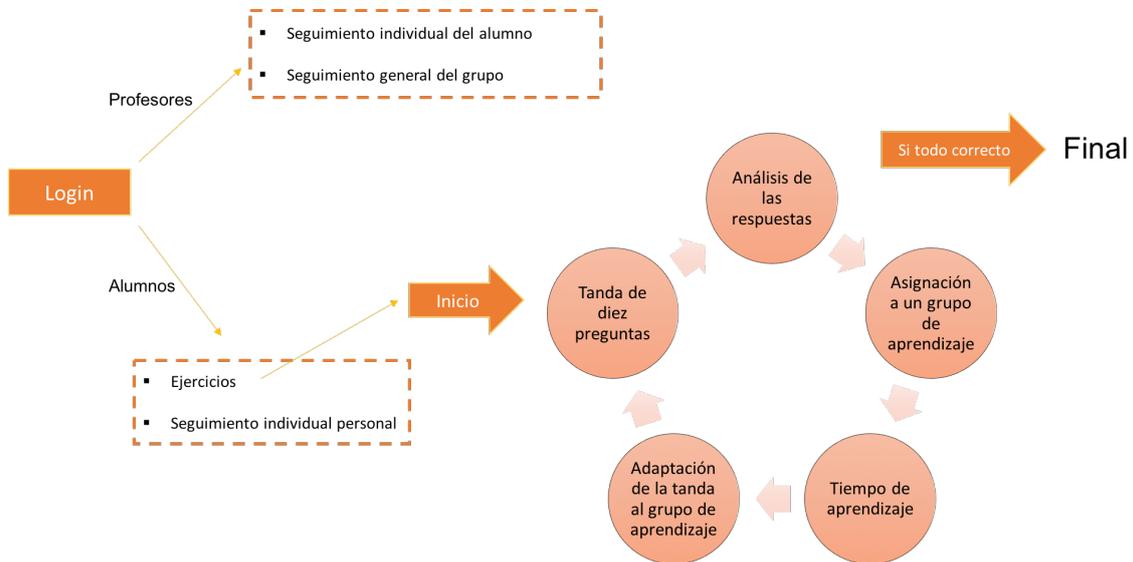
Una vez pasado esta etapa de aprendizaje, se podrá pasar a la nueva tanda de preguntas. En esta tanda, como se ha explicado anteriormente, se pondrá a prueba el aprendizaje que el alumno acaba de hacer sobre los conceptos que no había entendido en un primer tiempo.

Para no eternizar el alumno en un bucle del que no podrá salir si no es capaz de entender los conceptos, se ha definido un máximo de tres intentos. Tras estos tres intentos, se propondrá una tutoría con el profesor para que el problema de aprendizaje se pueda tratar de manera más personal. En el caso en el que el alumno agote los tres intentos, no se tratará de una pérdida de tiempo el haber realizado los ejercicios, si no que el tutor virtual habrá encontrado los conceptos que el alumno no termina de entender y el profesor podrá tratar el problema sabiendo exactamente hacia donde debe enfocar su tutoría.

Una vez implementado este proceso, se ha decidido hacer más útil el sistema añadiendo una etapa de autenticación. A través de la autenticación se puede saber qué alumno ha realizado qué ejercicios y cuál ha sido su respuesta ante ellos. Esto abre un gran abanico de posibilidades dado que, al identificar al alumno, se puede obtener un historial de su evolución y de cuales han sido sus errores más recurrentes.

Al poder identificar a los usuarios, se pueden diferenciar a los profesores y a los alumnos. Por ello se han diseñado dos grupos: Profesores y Alumnos. Los alumnos podrán acceder al sistema definido anteriormente y podrán consultar su evolución en su página personal. Los profesores podrán acceder a una plataforma diferente, en la que podrán observar la evolución individual de cada uno de los alumnos de su clase, así como la tendencia general del grupo. Este análisis de tendencias podría ayudar al profesor a determinar qué medidas se podrían tomar con un alumno o grupo de alumnos determinado. Si hay alumnos que están sistemáticamente por debajo de la media se les podrá ofrecer una ayuda complementaria a las clases.

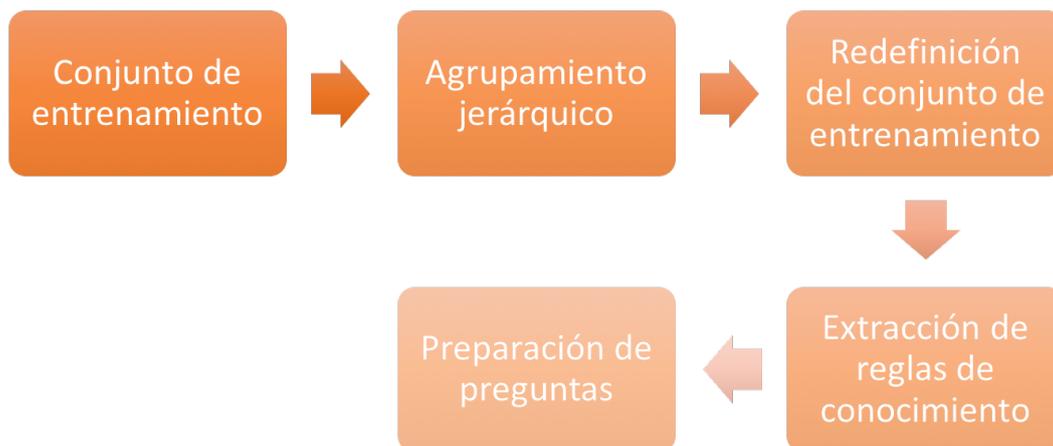
El proceso final quedaría tal y cómo se muestra en la Ilustración 7.



*Ilustración 7: Proceso del sistema final*

### 5.1.1 ELEMENTOS DEL SISTEMA

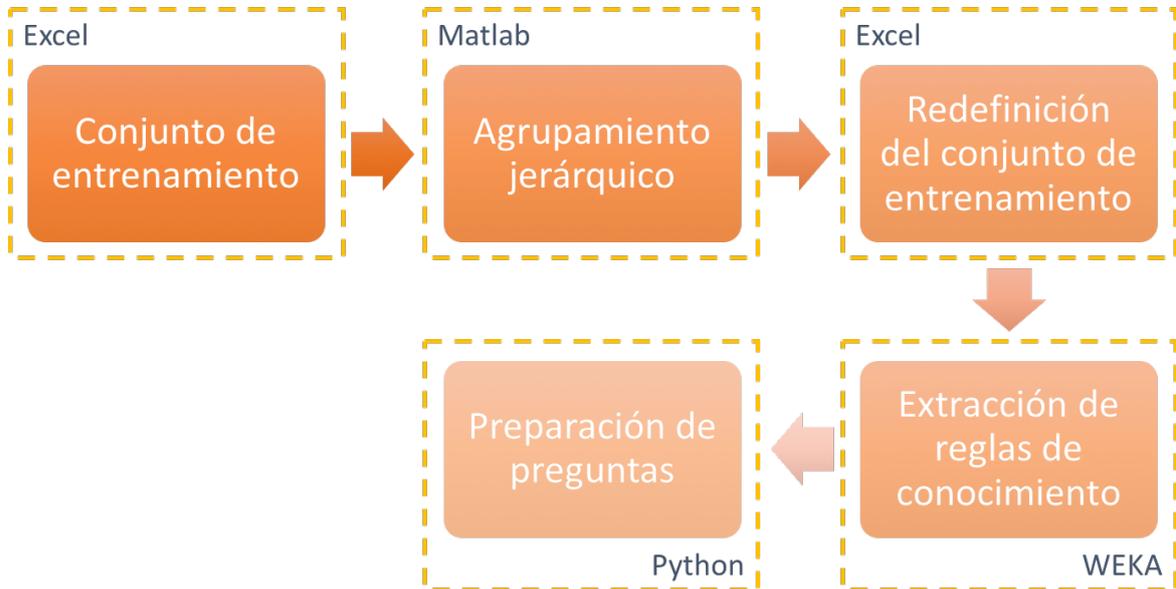
El diseño de esta plataforma está constituido por varios componentes. Algunos de ellos han sido trabajos previos a la implementación del sistema y otros forman parte de la interacción con la aplicación y su uso. Cada uno de estos elementos serán definidos en profundidad a lo largo de los siguientes apartados.



*Ilustración 8: Elementos previos al funcionamiento del sistema*

- Conjunto de entrenamiento: Conjunto de datos que alimenta el agrupamiento jerárquico, se trata del conjunto de datos que se ha utilizado para el estudio del perfil de los alumnos.
- Agrupamiento jerárquico: Divide el conjunto de entrenamiento en grupos de alumnos que tienen carencias o fallos similares en los verbos modales. La definición concreta de estos grupos se realiza en los siguientes apartados.
- Redefinición del conjunto de entrenamiento en otro que incluye el grupo al que pertenece cada alumno.
- Extracción de reglas de conocimiento con un árbol de decisión a partir del anterior componente. Estas reglas indican cuales son las características en cuanto a errores en verbos modales que califican o definen los grupos antes apreciados en los alumnos. En definitiva, se convierte el conjunto de entrenamiento en un conjunto de características de fallos catalogadas por grupos. Esto constituye la base de conocimiento.
- Preparación de preguntas de test o refuerzo: Conjunto de baterías de preguntas generadas de acuerdo a una estrategia según el grupo al que pertenece un usuario.

Para cada uno de estos elementos se ha utilizado una herramienta diferente, tal y como se puede observar en la Ilustración 9. Como se ha comentado anteriormente, a lo largo de los siguientes apartados se proporcionarán más detalles de cada uno de estos elementos.



*Ilustración 9: Elementos previos al funcionamiento del sistema y herramientas utilizadas*

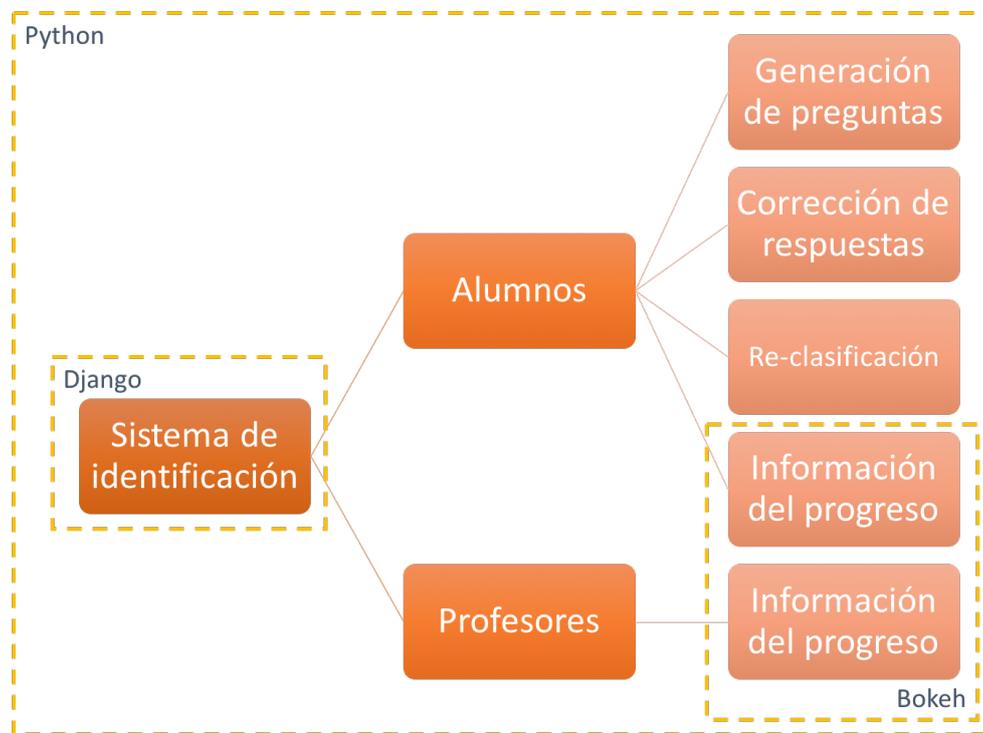
Del mismo modo, también se han definido componentes que trabajan en tiempo real a través de la aplicación, interactuando con alumno o profesor.



*Ilustración 10: Componentes de interacción con el sistema*

- Credenciales e identificación: En este punto se realiza una comprobación de si el usuario es un alumno o un profesor, y de esta manera poder ofrecer una interfaz adaptada a cada uno de ellos.
- Si procede, lectura de la base de conocimientos para generar preguntas. Esto se realiza en el caso de los alumnos.
- Corrección y comprobación de respuestas: Componente necesario para poder categorizar a los alumnos en función de las respuestas.
- Re-clasificación del alumno (hasta tres veces): Dependiendo del progreso del alumno se le asignará a un grupo o a otro, esto se evaluará cada vez que el alumno se vea enfrentado a una batería de preguntas.
- Componente informativo de progreso: Tanto profesor como alumno podrán acceder a un informe sobre la evaluación del alumno y sobre los errores que ha cometido en cada uno de sus ejercicios.

Estos últimos componentes se han realizado con Python, y dependiendo del componente se ha utilizado Django o librerías específicas de Python. Un detalle de las herramientas utilizadas para la definición de estos componentes se puede ver en la Ilustración 11.



*Ilustración 11: Componentes de interacción con el sistema y herramientas utilizadas*

## **5.2 ADQUISICIÓN Y GENERACIÓN DEL CONOCIMIENTO DEL SISTEMA**

Gracias a la ayuda de la experta, se ha tenido acceso exámenes y ejercicios realizados por los alumnos de las clases a las que la profesora imparte inglés. Estos exámenes se han tratado de forma anónima y confidencial para que la identidad de los alumnos no trascienda, dado que es irrelevante para el desarrollo del sistema.

Se han tomado como modelo los ejercicios en los cuales se presenta una frase con un blanco y se debe rellenar ese blanco con un modal concreto. Un ejemplo sería el siguiente:

*I \_\_\_\_\_ go to the concert tonight, but I'm still not sure.*

Se ha determinado, junto con la experta, que este tipo de preguntas sería fácilmente adaptable a un sistema automático. También se ha barajado la posibilidad de utilizar ejercicios en los cuales se propone una frase y se debe transformar la misma incluyendo verbos modales. No se ha deseado continuar con este tipo de ejercicio por ahora, pero podría ser objeto de trabajos futuros. Este conjunto de casos es lo que se ha usado como conjunto de entrenamiento para luego extraer del mismo conocimiento de manera automática.

Para generar el conocimiento que contendrá la base de conocimiento del sistema se ha contado con la información de 74 alumnos de niveles diferentes (entre 16 y 18 años) y de ejercicios diferentes. Esto se debe a que los ejercicios propuestos en un examen son distintos del anterior. Esto ha supuesto algún problema a la hora de clasificar a los alumnos, pero se explicará más adelante.

Para crear dicho conjunto de entrenamiento se ha utilizado Microsoft Excel, ya que es una aplicación muy útil para crear sencillas bases de datos estructuradas. En las columnas se ha optado por describir cual era la respuesta correcta a los ejercicios y para cada alumno se ha ido anotando si la respuesta que había dado era correcta o incorrecta. En caso de ser incorrecto, se ha apuntado también cuál ha sido el fallo.

Para poder analizar la base de datos, cada tipo de respuesta se ha codificado. Por ejemplo 0 quiere decir que la respuesta ha sido correcta, pero el código 20 quiere decir que la respuesta ha sido incorrecta y que en vez de responder lo que se esperaba se ha respondido *would*. En total hay 41 códigos en los que se tiene en cuenta también si realmente no se han entendido los verbos modales globalmente. Los códigos se pueden consultar en el Anexo A – Códigos de respuesta.

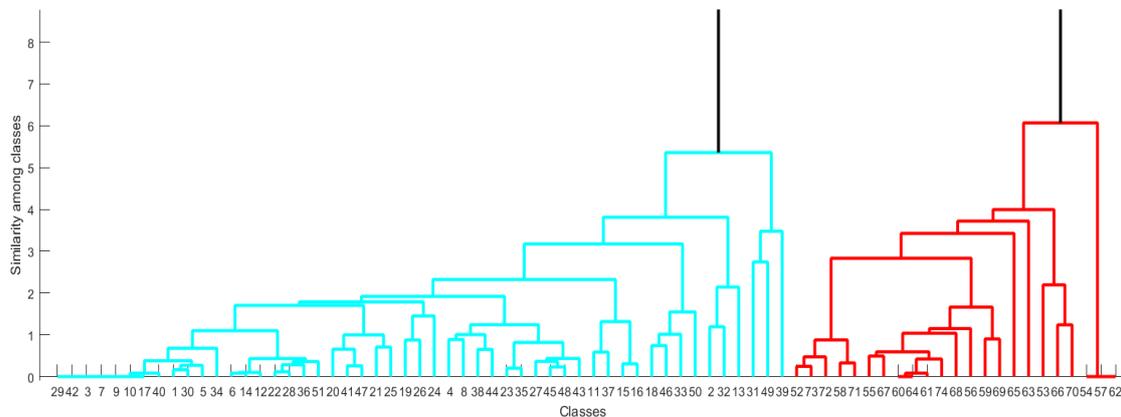
Esta codificación ha sido muy útil para identificar el perfil de los alumnos, tal y como se explicará más adelante, pero también se ha debido tratar el conjunto de entrenamiento para poder obtener las reglas de inferencia del sistema experto.

Este conocimiento adquirido ha sido el pilar más importante del proyecto, dado que sin estos datos el sistema no hubiera podido desarrollarse. El hecho de contar con datos reales sobre el aprendizaje de los alumnos es muy valioso y siempre que ha sido necesario se ha vuelto a la base de datos original para extraer información clave para el sistema. Todo esto ha sido posible gracias a la experta que ha prestado su ayuda al proceso y sin la cual este proyecto no hubiera sido posible.

### **5.2.1 DEFINICIÓN DEL PERFIL DE LOS ALUMNOS**

El perfil de los alumnos se ha determinado a partir del conjunto de entrenamiento mencionado anteriormente. Como se ha explicado en el apartado previo, el hecho de tener dos ejercicios diferentes en la base de conocimiento ha dificultado el proceso de identificación de perfiles. Para ello se ha creado un código especial, el código 500, que quiere decir que ese ejercicio no se ha realizado por ese alumno en concreto.

Se ha utilizado la técnica de agrupamiento jerárquico para poder identificar los perfiles o grupos de alumnos de características similares en cuanto a las respuestas que han dado a diferentes tests. Para llevar a cabo este agrupamiento se ha usado Matlab y las funciones encargadas de ello. Los resultados se han mostrado de manera gráfica con la función Dendogram, que representa el dendrograma de resultados del estudio. El dendrograma es un árbol basado en la similitud de las columnas del conjunto de datos de entrada que se introducen en la función. El conjunto de entrenamiento ha sido transpuesto, con el comando Transponer de Excel, para poder comparar las columnas de la manera que se necesitaba para obtener el perfil de alumnos. El resultado de la función Dendogram de Matlab se muestra en la Ilustración 12.



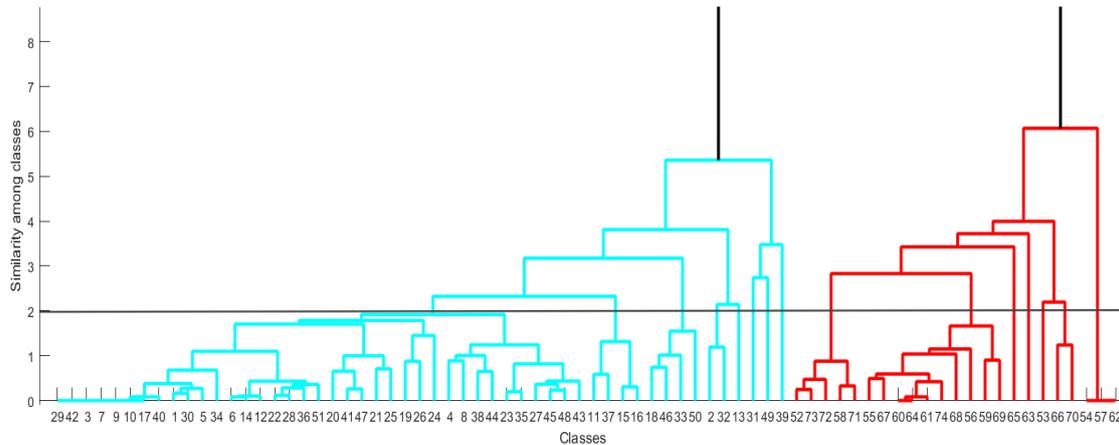
*Ilustración 12: Perfil de los alumnos*

En la Ilustración 12 se observa que la similitud entre grupos de perfiles es bastante alta. Hay que tener en cuenta que la función Dendrogram crea un gráfico que mide  $(100-x)$  en similitud. Esto quiere decir que cuando hay un 5% de *Similarity among classes*, hay un 95% de similitud entre el perfil de los alumnos. Esto puede llevar a confusión, por lo que es importante tenerlo claro.

En el dendograma se puede observar claramente la diferencia entre los dos ejercicios que se han comentado antes. Los alumnos de la parte azul son aquellos que han realizado el primer ejercicio y los de la parte roja son aquellos que realizaron el segundo ejercicio.

Dada la similitud entre los alumnos, en un principio se decidió crear Grupos de Aprendizaje con un 99% de similitud entre los alumnos. Al tener un porcentaje tan elevado de similitud, el tratamiento de los alumnos sería muy personalizado y adaptado a cada uno de ellos. Desafortunadamente, después de tratar el sistema a partir con este nivel de similitud, se observó que el sistema era demasiado complejo como para seguir por este camino.

Se decidió por tanto disminuir el grado de similitud de los alumnos al 98%. Podría parecer que el cambio del 99% al 98% no fuera muy determinante, pero dada la estructura del dendograma resultó muy conveniente.



*Ilustración 13: Perfil de los alumnos con un grado de similitud del 98%*

En la Ilustración 13 se puede observar como la complejidad de los grupos del dendograma disminuye cuando tomamos 98% como corte. Los grupos al 98% son menos numerosos y su tratamiento es más sencillo.

### **5.2.2 CLASIFICACIÓN EN GRUPOS DE APRENDIZAJE**

La clasificación en grupos de aprendizaje se hace a partir de la línea gris trazada en la Ilustración 13. Como explicado en el apartado anterior, este corte se hace para que los grupos de aprendizaje cuenten con alumnos que tengan perfiles con un 98% de similitud.

El funcionamiento del dendograma es el siguiente. Se toman grupos a partir de las líneas verticales que cortan con la línea horizontal dibujada. De esta manera el primer grupo contará con los alumnos 1, 3, 4, 5, 6, 7, 8, 9, 10, 12, 14, 17, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 34, 35, 36, 38, 40, 41, 42, 43, 44, 45, 47, 48 y 51 (el grupo 1 de aprendizaje es el más numeroso).

En el ANEXO B – Grupos de Aprendizaje, se puede observar la repartición de los alumnos estudiados en cada uno de los grupos.

Una vez distribuidos los alumnos en Grupos de Aprendizaje, es importante entender cuáles han sido los errores de los alumnos que los constituyen. Para entender esto, se ha retomado la base de conocimiento y se ha ido mirando, grupo por grupo, cuáles han sido sus errores comunes y se ha determinado un patrón para cada uno de ellos. Los errores más aislados no se han tenido en cuenta.

La siguiente tabla es un ejemplo del proceso que se ha seguido, se toma el Grupo 2 que contiene a los alumnos 11, 15, 16 y 37.

*Tabla 2: Ejemplo de análisis de los Grupos de Aprendizaje*

	<b>might/ may</b>	<b>mustn't</b>	<b>can't</b>	<b>would / could / can</b>	<b>didn't have to</b>	<b>should</b>	<b>don't need</b>	<b>must have</b>	<b>Class</b>
11	0	0	0	0	36	-	-	-	G2
15	2	0	1	15	24	-	-	-	G2
16	1	0	4	15	0	-	-	-	G2
37	1	0	0	7	35	-	-	-	G2

Tal y como se ve en la Tabla 2, este grupo se caracteriza claramente por entender el concepto de *mustn't*. Generalmente se encuentran dificultades en *might/may*, que confunden con conceptos del grupo *can* (*can* y *can't*, específicamente); en *would/could/can*, que confunden con *shall* y *may*; y *didn't have* no ha sido respondido correctamente en la mayoría de los casos, se ha respondido *couldn't have*, *should have* y *shouldn't have*.

A este grupo se le ha recomendado un repaso de los conceptos *can*, *may*, *might*, *have*, *should* y *would*. Repasar estos conceptos se entiende como repasar presente positivo, presente negativo, pasado positivo y pasado negativo. Es decir, en el caso del grupo *may* se recomendaría repasar *may*, *may not*, *may have* y *may not have*.

Este análisis se ha hecho para cada uno de los nueve grupos que se han determinado al estudiar el perfil de los alumnos.

En el ANEXO C – Grupos de Aprendizaje y Recomendaciones, se muestra qué áreas deben mejorar cada uno de los grupos.

### 5.3 DEFINICIÓN DE LA BASE DE CONOCIMIENTOS

Las reglas de inferencia sirven para clasificar a los alumnos en función de las respuestas que han dado. Constituyen el algoritmo base para desarrollar el tutor experto virtual, ya que a partir de la clasificación que se haga se podrán hacer unas recomendaciones u otras.

Para obtener dichas reglas de inferencia se ha utilizado la plataforma WEKA. Esta plataforma es muy útil para elaborar árboles de decisión y reglas de inferencia a partir de los datos insertados en el programa.

El primer paso para poder utilizar WEKA es tratar la base de datos para que la herramienta pueda entender y tratar la información. A la base original se le ha añadido el grupo al cual el alumno pertenece, ya que es la clase que WEKA va a utilizar y testar.

*Tabla 3: Extracto de la tabla de datos introducida en WEKA*

	Might / may	mustn't	can't	would / could / can	didn't have to	should	don't need	must have	Class
1	C	C	C	C	I	OE	OE	OE	G1
2	C	I	NA	C	I	OE	OE	OE	G4
3	C	C	C	C	C	OE	OE	OE	G1
4	I	I	I	C	I	OE	OE	OE	G1
5	C	C	I	C	I	OE	OE	OE	G1
6	C	C	C	C	I	OE	OE	OE	G1

7	C	C	C	C	C	OE	OE	OE	G1
8	I	I	I	C	I	OE	OE	OE	G1
9	C	C	C	C	C	OE	OE	OE	G1
10	C	C	C	C	C	OE	OE	OE	G1
11	C	C	C	C	I	OE	OE	OE	G2
12	C	I	C	C	I	OE	OE	OE	G1
13	C	C	I	C	I	OE	OE	OE	G4
14	C	C	C	C	I	OE	OE	OE	G1
15	I	C	I	I	I	OE	OE	OE	G2

La Tabla 3 es un extracto de la tabla que se ha introducido en WEKA para su tratamiento.

- C: La respuesta es correcta
- I: La respuesta es incorrecta
- OE: Este ejercicio no se ha realizado

Como se puede observar, en este paso se ha perdido la posibilidad de conocer cuál ha sido el motivo del fallo. Esta característica se recuperará más adelante durante la etapa de desarrollo, pero para el funcionamiento de WEKA no se ha considerado necesario. Lo que se va a mirar es cómo el fallo en alguno de los conceptos estudiados influye en la pertenencia a un Grupo de Aprendizaje. Dado que los Grupos de Aprendizaje se han asignado de manera manual, WEKA va a ayudar a comprender si esta asignación es correcta y tiene sentido.

El archivo que toma WEKA es un .arff, y este debe tomar un formato particular para que la herramienta pueda entenderlo, explicado anteriormente. A continuación se muestra un extracto del archivo:

@relation alumnos

@attribute mightmay {C,I,NA,OE}

@attribute mustnt {C,I,NA,OE}

@attribute cant {C,I,NA,OE}

@attribute wouldcouldcan {C,I,NA,OE}

@attribute didnthaveto {C,I,NA,OE}

@attribute should {C,I,NA,OE}

@attribute dontneed {C,I,NA,OE}

@attribute musthave {C,I,NA,OE}

@attribute Clase {G1,G2,G3,G4,G5,G6,G7,G8,G9}

@data

C, C, C, C, I, OE, OE, OE, G1

C, I, NA, C, I, OE, OE, OE, G4

C, C, C, C, C, OE, OE, OE, G1

I, I, I, C, I, OE, OE, OE, G1

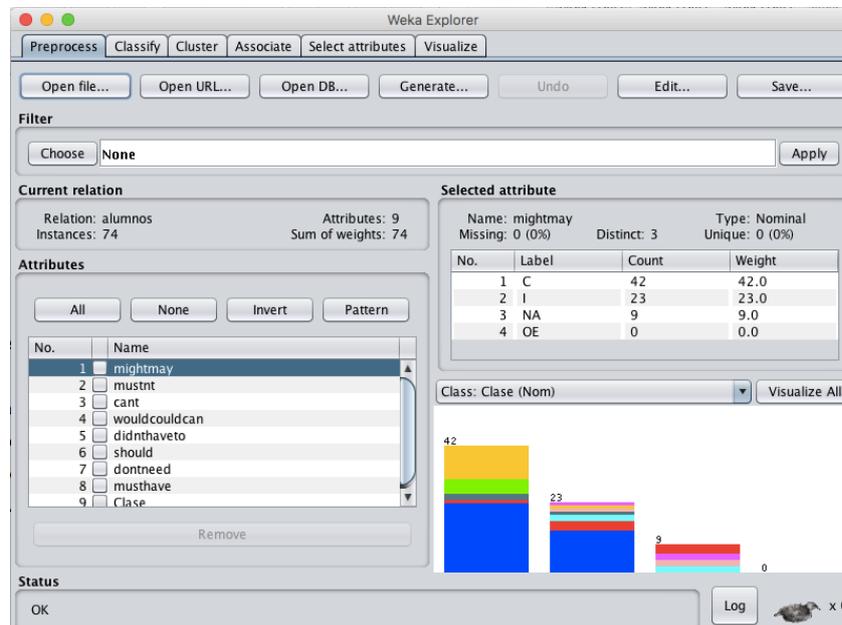
C, C, I, C, I, OE, OE, OE, G1

C, C, C, C, I, OE, OE, OE, G1

C, C, C, C, C, OE, OE, OE, G1

En la Ilustración 14, se observa la interfaz de WEKA tras haber añadido los datos a la misma.

Las pestañas que se han utilizado son la de *Classify* y *Associate*.



*Ilustración 14: Interfaz de WEKA tras añadir los datos a la plataforma*

*Associate* sirve para inferir las reglas que hacen que unos datos obtengan una clase u otra. Es cierto que en este caso esta funcionalidad no ha sido muy útil. Esto se debe al hecho de tener alumnos que no han realizado alguno de los ejercicios ha falseado estas reglas.

Es por ello que se ha decidido partir de un árbol de decisión para poder obtener las reglas de inferencia. *Classify* es la pestaña que tiene por función la de crear árboles de decisión a partir de los datos introducidos.

```

musthave = C: G1 (0.0)
musthave = I
|   mightmay = C: G7 (16.0/5.0)
|   mightmay = I: G8 (1.0)
|   mightmay = NA: G8 (2.0)
|   mightmay = OE: G7 (0.0)
musthave = NA: G9 (4.0/1.0)
musthave = OE
|   cant = C
|   |   mightmay = C: G1 (16.0/1.0)
|   |   mightmay = I: G1 (4.0/2.0)
|   |   mightmay = NA: G3 (2.0)
|   |   mightmay = OE: G1 (0.0)
|   cant = I: G1 (24.0/4.0)
|   cant = NA
|   |   mustnt = C: G5 (0.0)
|   |   mustnt = I: G4 (3.0/1.0)
|   |   mustnt = NA: G5 (2.0)
|   |   mustnt = OE: G5 (0.0)
|   cant = OE: G1 (0.0)

```

*Ilustración 15: Extracto del árbol de decisión resultante*

En el árbol de decisión se observa que hay unos números junto al grupo en el que se ha clasificado cada una de las ramas. Esto se refiere a los casos que se han clasificado correctamente / y los que no se han clasificado correctamente respecto a la clasificación realizada manualmente.

En vista de los resultados obtenidos, se ha investigado cada uno de los casos que podían resultar conflictivos para determinar si el Grupo de Aprendizaje debía ser cambiado o si había una justificación a esa clasificación.

La combinación de las reglas proporcionadas por WEKA y del árbol de decisión se ha utilizado para poder determinar las reglas de inferencia del sistema. También se ha realizado un análisis de cada uno de los grupos para identificar patrones. Se ha considerado en esta etapa realizar un estudio lo más detallado posible, el cual se podrá simplificar en la etapa de desarrollo.

### 5.3.1 REGLAS DE INFERENCIA

El resultado del anterior análisis concluye en las siguientes reglas de inferencia:

R1: must have = 0  $\rightarrow$  G1

R2: must have < 39 && (might/may = 0 || might/may = 39)  $\rightarrow$  G7

R3: must have >= 39 && (might/may = 0 || might/may = 39)  $\rightarrow$  G6

R4: must have  $\diamond$  0 && must have  $\diamond$  39 && might/may  $\diamond$  0  $\rightarrow$  G8

R5: must have = 39 && should  $\diamond$  0  $\rightarrow$  G9

R6: must have = 39 && should = 0  $\rightarrow$  G7

R7: can't = 0 && might/may < 39  $\rightarrow$  G1

R8: can't = 0 && (might/may = 0 || might/may  $\diamond$  39) && would/could/can  $\diamond$  0  $\rightarrow$  G2

R9: can't = 0 && might/may >= 39  $\rightarrow$  G3

R10: can't  $\diamond$  0 && can't < 39  $\rightarrow$  G1

R11: can't  $\diamond$  0 && might/may >= 39  $\rightarrow$  G3

R12: can't  $\diamond$  0 && might/may < 39 && would/could/can > 14 && would/could/can < 17  
 $\rightarrow$  G2

R13: can't  $\diamond 0$  && can't  $\geq 39 \rightarrow G4$

R14: can't = 39 && (mustn't = 0 || mustn't = 39)  $\rightarrow G5$

R15: can't = 39 && mustn't  $\diamond 0$  && mustn't  $\diamond 39 \rightarrow G4$

R16: can't = 39 && mustn't  $\diamond 0$  && mustn't  $\diamond 39$  && didn't have to  $< 39 \rightarrow G5$

R17: didn't have to = 0  $\rightarrow$  mustn't = 0

R18: might/may  $\diamond 0$  && might/may  $\diamond 39$  && can't  $\diamond 0$  && can't  $\diamond 39$  &&  
would/could/can = 0  $\rightarrow G1$

R19: can't  $\diamond 0$  && can't  $\diamond 39$  && would/could/can = 0  $\rightarrow$  didn't have to  $> 0$  && didn't  
have to  $< 39$  && didn't have to  $> 41$

R20: might/may = 0 && can't = 0  $\rightarrow$  would/could/can = 0

R21: might/may = 0 && can't = 0 && would/could/can = 0  $\rightarrow G1$

## **5.4 IMPLEMENTACIÓN**

En estos apartados se explicará cómo se ha desarrollado la plataforma y cómo se han utilizado las distintas herramientas durante dicho desarrollo. La parte de implementación es la que más tiempo y esfuerzo se ha invertido. La parte visual de una aplicación es la que la hace atractiva para su uso y la que finalmente la hace útil, es por eso que se ha considerado importante el hecho de contar con una interfaz con la cual el usuario pueda interactuar fácilmente. Además, desde el punto de vista del desarrollador, siempre es útil poder ir testando la aplicación a medida que se va construyendo.

### 5.4.1 GENERACIÓN ALEATORIA DE FRASES

Para el desarrollo de esta aplicación de soporte para el aprendizaje del inglés, se ha pensado que sería interesante que se pudieran generar frases aleatorias cada vez que el programa se pone a funcionar. Con esto se busca evitar que los alumnos acaben por aprenderse las respuestas de memoria, siendo cada ejercicio un nuevo reto que poder resolver. Además, con este método se evita que los alumnos se puedan copiar entre sí, ya que habría pocas posibilidades de que dos alumnos obtuvieran las mismas preguntas.

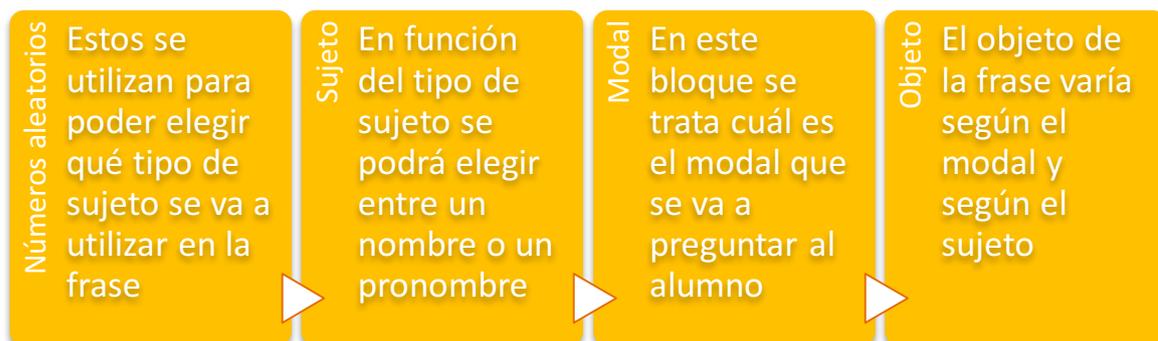
Para la creación de este algoritmo, se ha decidido comenzar con un prototipo diseñado en un lenguaje conocido por la autora de la tesis. Dicho prototipo se ha desarrollado en Excel con la ayuda del lenguaje de programación Visual Basic. Se ha usado para probar que la generación aleatoria de frases se podría llevar a cabo en un lenguaje más avanzado.

Básicamente el algoritmo se basa en cuatro bloques divididos de la siguiente manera:

- Número aleatorio: Se trata de un set de números del uno al cuatro, siendo cada uno de ellos un tipo de sujeto.
  - 1: *I*
  - 2: *You, we* o *they*
  - 3: *She* o nombre femenino
  - 4: *He* o nombre masculino
- Sujeto: Una vez se ha decidido el tipo de sujeto, en los casos dos, tres y cuatro, se elige qué se va a utilizar como tal. Se han propuesto una parrilla de nombres masculinos y femeninos en cada uno para los casos tres y cuatro. El hecho de poner optar a

nombres en vez de simplemente a los pronombres cómo sujeto hace que la generación de frases tenga un toque más personalizado y original. Para el caso dos, se deberá elegir entre *you*, *we* o *they*.

- **Modal:** En este bloque se elige de manera aleatoria cuál es el modal que va a ser objeto de la pregunta para el alumno. Se divide en dos, dependiendo del tipo de sujeto que se haya obtenido dado que algunos verbos tienen variantes para la tercera persona del singular.
  - Modal básico
  - Modal para tercera persona
- **Objeto:** Finalmente, en función del modal asignado, se propone una parrilla de continuaciones para la frase que se está definiendo. En función del sujeto, esta terminación puede variar y es lo que hace que la frase tenga sentido.



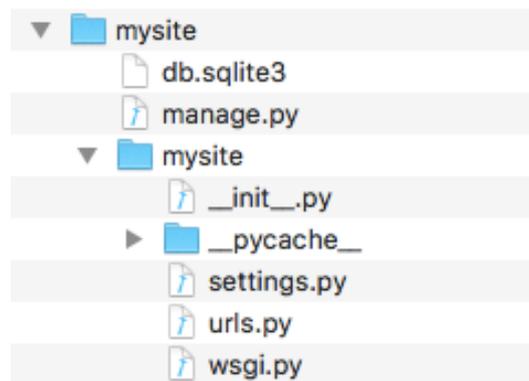
*Ilustración 16: Proceso de generación aleatoria de frases*

Una vez realizado el prototipo y probado su funcionamiento, se ha pasado a desarrollarlo en un lenguaje más apropiado para la aplicación en su conjunto. Se ha decidido optar por

Python, teniendo en cuenta la API que ofrece para sistemas analíticos. Dado que este proyecto está basado en el análisis de datos, se ha considerado como la opción más adecuada.

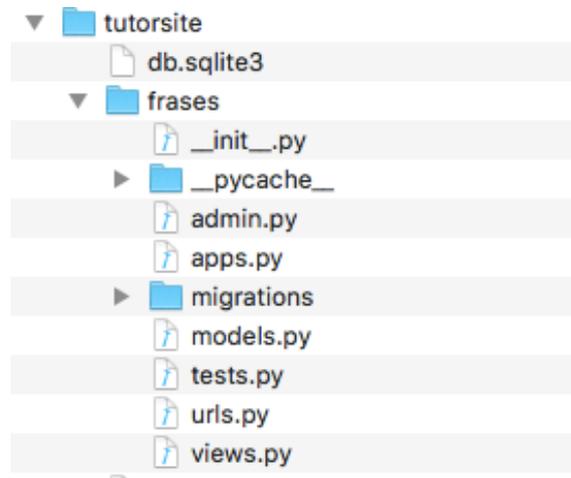
Además, dado que se ha decidido desarrollar una plataforma web para que los usuarios puedan acceder a la aplicación, se ha utilizado Django. Se trata de un framework de Python muy útil para el desarrollo de aplicaciones web, ya que define muy bien la utilidad de cada módulo.

Para hacerse una idea de lo que significa utilizar Django al comenzar un nuevo proyecto, se puede observar la Ilustración 17. Este es el aspecto del directorio de un proyecto nada más ser creado a través de Django. Un proyecto puede contener múltiples aplicaciones, y en el desarrollo de este proyecto se ha hecho uso de este punto fuerte que tiene Django. Como dicho anteriormente, cada uno de los documentos y carpetas tienen una utilidad definida por el propio framework, por lo que la creación de una aplicación web es más sencilla.



*Ilustración 17: Ejemplo de creación de un proyecto con Django*

Como se ha explicado anteriormente, Django ofrece la posibilidad de crear distintas aplicaciones dentro de un mismo proyecto. Esto es muy útil cuando se quieren tener distintas funcionalidades y se quiere que estas sean independientes.



*Ilustración 18: Creación de una aplicación dentro de un proyecto de Django*

En el caso de esta tesis, se ha creado un proyecto de Django, que se ha denominado *tutorsite*. Dentro de este proyecto se ha definido una aplicación que se ocupa de generar frases, se ha denominado *frases*. Como se puede ver en la Ilustración 18, también al crear una aplicación se generan carpetas y archivos con funcionalidades específicas. Los archivos más relevantes para la creación de esta aplicación han sido *models.py* y *views.py*.

*Models.py* es el archivo en el cual se diseña la base de datos, la misma que se ha explicado en apartados anteriores. Este archivo se encarga de generar automáticamente las tablas necesarias en la base de datos que Django utiliza por defecto, SQLite. Para la realización de la aplicación no es necesario entrar en la base de datos como tal.

```

1  from django.db import models
2
3  # Create your models here.
4
5  class Numero(models.Model):
6      numero = models.IntegerField()
7      def numeros(self):
8          return self.numero
9
10 class Sujeto_i(models.Model):
11     sujeto_i = models.CharField(max_length=100)
12     def __str__(self):
13         return self.sujeto_i
14
15 class Sujeto_you(models.Model):
16     sujeto_you = models.CharField(max_length=100)
17     def __str__(self):
18         return self.sujeto_you
19
20 class Sujeto_he(models.Model):
21     sujeto_he = models.CharField(max_length=100)
22     def __str__(self):
23         return self.sujeto_he
24
25 class Sujeto_she(models.Model):
26     sujeto_she = models.CharField(max_length=100)
27     def __str__(self):
28         return self.sujeto_she
29

```

Ilustración 19: Extracto del archivo Models.py de la aplicación de frases

Django ofrece la posibilidad de acceder a la base de datos de un modo más intuitivo a través de su interfaz de administrador. Esta interfaz tiene como ventajas que es muy *user friendly* y no es necesario el conocimiento de SQL para poder modificar o acceder a los datos. Para la realización de este proyecto, este ha sido el método utilizado para poblar la base de datos.

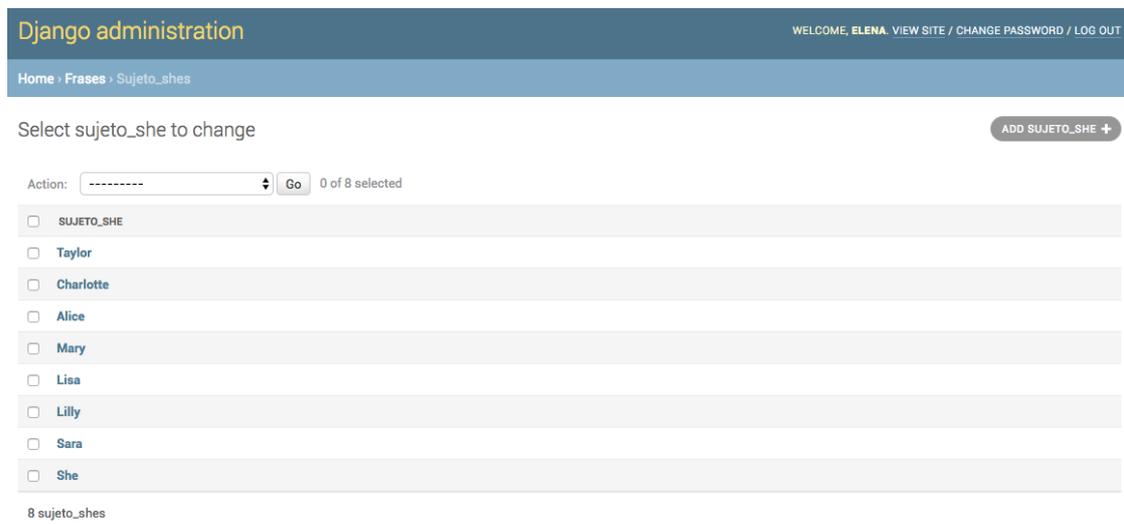


Ilustración 20: Interfaz de administrador de Django

Otra manera para poblar la base de datos de la aplicación es a través del Shell que proporciona Python. Sin embargo, este método es más lento y mucho menos intuitivo.

El otro archivo importante para el desarrollo de una aplicación en Django es *Views.py*. En este archivo es donde se va a desarrollar el programa como tal. Originariamente sirve para crear las vistas de una aplicación web, pero esta aplicación simplemente va a servir de apoyo al proyecto general y no va a tener una vista propia. En este caso *views.py* se ha utilizado para generar las funciones y la lógica necesaria para crear las frases de forma aleatoria.

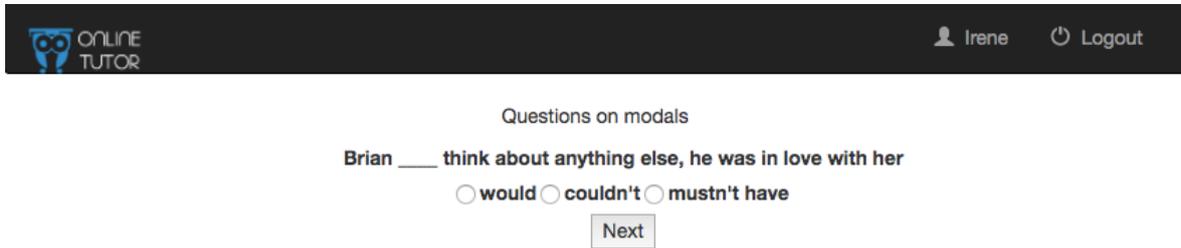
La lógica que se ha utilizado para definir el programa en Python ha sido la misma que para el prototipo en Excel. Con esta aplicación, el alumno podrá obtener retos distintos cada vez que acceda a la plataforma para seguir aprendiendo. La cantidad de posibilidades depende de cuánto se pueble la base de datos, para este desarrollo se han utilizado una media de 2,5 objetos para cada tipo de modal, para cada tipo de sujeto. Lo que ofrece un número de 340 posibilidades distintas cada vez que se genera una nueva frase.

Las frases que aparecen en esta aplicación están basadas en ejemplos proporcionados por [23] y [24].

### **5.4.2 GENERACIÓN DE EJERCICIOS**

El ejercicio cómo tal se ha decidido tratar como una aplicación separada en Django. Se considera ejercicio a una tanda de diez preguntas en las que se pone a prueba el nivel del alumno en cuanto a los verbos modales.

En cada pregunta, cada frase generada, se va a esconder el modal (la solución) y se van a proponer tres posibilidades. El alumno deberá escoger una de las tres posibilidades y pasar a la siguiente pregunta.



*Ilustración 21: Vista de un ejercicio de la aplicación*

Es necesario explicar que el alumno dispone de tres oportunidades para realizar los ejercicios correctamente. Pasadas dichas oportunidades, se propondrá que el profesor se ocupe del alumno dado que la aplicación no es capaz de ayudarle a comprender los conceptos con los cuales dicho alumno tiene dificultades. No se pretende que el alumno siga en un bucle infinito intentando acertar las respuestas, si no que la aplicación sirva para su desarrollo y aprendizaje.

El código de estos ejercicios se basa en hacer una llamada a la aplicación que genera las frases. Al mismo tiempo se recupera la respuesta correcta a la pregunta y se generan dos otras posibilidades. Estas tres posibles respuestas se van a mezclar de forma aleatoria para que la respuesta correcta se sitúe cada vez en una posición distinta. Una vez el alumno escoge una de las opciones y se pulsa el botón de siguiente, la respuesta dada se guarda y se compara con la respuesta correcta.

Cuando el alumno acaba el ejercicio, llega a una página en la que se hace un resumen de sus fallos y se le proporcionan distintas opciones.



Ilustración 22: Vista de la aplicación al acabar un ejercicio

Estas opciones se van a comentar en los siguientes apartados del capítulo.

### 5.4.3 CLASIFICACIÓN EN GRUPOS DE APRENDIZAJE

El análisis de las respuestas que el alumno ha ido dando durante el ejercicio finaliza con su clasificación en alguno de los Grupos de Aprendizaje mencionados en apartados anteriores. Como se ha ido explicando a lo largo de este documento, esta clasificación en grupos permite que el alumno reciba una atención personalizada por parte de la aplicación.

La manera de clasificar a los alumnos se hace a través de un algoritmo generado a partir de las reglas de inferencia previamente definidas. Como se ha podido observar, las reglas de inferencia son muy detalladas y se podría dar en caso de que ninguna de las preguntas que se han propuesto al alumno corresponda con alguna de las reglas. Por esta razón se ha decidido realizar una versión de las reglas de inferencia a alto nivel. Esto quiere decir que en vez de mirar si el alumno ha realizado un ejercicio que incluya *can't*, se va a mirar si alguna de las preguntas respondidas por el alumno pertenece al grupo de *can*, que incluye la forma negativa y pasada.

Además, en vez de mirar el tipo de error cometido, se ha decidido analizar si la respuesta ha sido correcta o incorrecta. Esto hace que el sistema de clasificación sea más sencillo de implementar, pero también a la hora de clasificar a los alumnos.

```
2806     if "would" in errores_c or "could" in errores_c or "can" in errores_c:  
2807         grupo = "grupo2"  
2808     elif "must" in errores_c and "can" in errores_c:  
2809         grupo = "grupo4"  
2810     elif ("might" in errores_c or "may" in errores_c) and "can" in errores_c:  
2811         grupo = "grupo3"
```

*Ilustración 23: Extracto del código del algoritmo de clasificación*

Esta clasificación en grupos va a condicionar el tiempo de aprendizaje y la siguiente ronda de preguntas que se le propongan al alumno.

#### **5.4.3.1 Tiempo de aprendizaje**

El objetivo de la aplicación es que el alumno sea capaz de comprobar sus conocimientos y reforzar aquellos que tiene menos arraigados. Por ello, no se pretende que el alumno encadene rondas de preguntas sin pararse a pensar cuáles han sido los errores y cómo mejorar para la próxima ronda.

Si se vuelve a la Ilustración 22, se observa que se le proporciona al alumno la opción de ver cuáles han sido sus respuestas. El objetivo de esta vista de la aplicación es que el alumno vea cuáles han sido las preguntas que ha acertado, cuáles ha fallado, y que pueda reflexionar cuál sería la respuesta correcta.

Tanto en el caso de acertar o fallar una pregunta, no se proporciona al alumno ni la respuesta correcta ni la respuesta seleccionada. Se ha diseñado la vista de esta manera para evitar que el alumno aprenda de memoria los fallos y no reflexione en caso de que una pregunta similar surja en otra ronda.

ONLINE TUTOR Irene Logout

### See the answers

I \_\_\_ visited you, but I wanted to call first  
don't need  
mustn't  
would have

Sara \_\_\_ done it without you, you have been a great support.  
shouldn't  
couldn't have  
wouldn't

I \_\_\_ invite her to the party, she doesn't get along with Tim  
can't  
don't have  
shouldn't

Lucas \_\_\_ fallen off his bike again. He came home with an injured knee today.  
must have  
would  
don't have

Ilustración 24: Vista de la aplicación en la opción See the answers

Como se puede ver en la Ilustración 24, las preguntas que se han fallado se señalan en rojo. En un principio se pensó únicamente señalar las preguntas erróneas en este color, pero finalmente se ha decidido marcar las respuestas acertadas en verde para motivar al alumno y que no tenga una visión solamente “roja” de sus respuestas. También se pretende reforzar al alumno en las preguntas que se han contestado correctamente para que siga ese camino.

La segunda parte del tiempo de aprendizaje es más teórica y se trata de mostrar al alumno cuáles son los conceptos que debe revisar antes de comenzar una nueva ronda de preguntas.

**Can**  
We use *can* to make statements of what is...  
[Read More](#)

**Have to**  
*Have to* is used to express strong obligation...  
[Read More](#)

**May**  
We use *may* when we are not sure about something...  
[Read More](#)

**Might**  
We use *might* when we are not sure about something...  
[Read More](#)

**Should**  
*Should* is used to express a mild obligation...  
[Read More](#)

**Would**  
We use *would* as the past tense of *will*...  
[Read More](#)

[Try again !](#)

Ilustración 25: Vista de revisión de conceptos de la aplicación

Esta revisión de conceptos se adapta al Grupo de Aprendizaje en el cual se ha clasificado al alumno. En cada una de las opciones se propone aprender más sobre los conceptos que se considera que los alumnos de determinado grupo no terminan de comprender. El alumno es el encargado de tomarse el tiempo necesario para entender cada concepto. Una vez él considere que se ha completado el aprendizaje, puede volver a realizar otro ejercicio pulsando el botón *Try again!*.

La información mostrada en los el apartado *Review the topics*, y en cada una de las páginas de conceptos, ha sido proporcionada por [25].

### 5.4.3.2 Generación de preguntas adaptadas

Como se ha comentado anteriormente, el objetivo de la aplicación es poder clasificar a los alumnos en función de su aprendizaje y poder tratar a cada uno de ellos en función de los errores que cometen. Para poder realizar esta adaptación es necesario que el algoritmo de generación de frases sea capaz de responder ante el tipo de grupo en el que se ha clasificado a cada alumno.

Esta respuesta se basa en proporcionar frases que intentar reforzar el conocimiento que el alumno todavía no ha adquirido. Es decir, si un alumno está clasificado en el Grupo de Aprendizaje número tres, esto quiere decir que se van a generar frases que contengan los conceptos *have*, *may* y *might* (y sus formas negativas y pasadas). Esta funcionalidad es la que hace que el algoritmo sea capaz de adaptarse a cada alumno y proporcione retos adecuados al nivel del mismo.

Como se puede suponer, los conceptos que se propondrán tras clasificar en un grupo a un alumno coinciden con los conceptos que se han propuesto para revisión.

```
171 def genera_pregunta(request):
172
173     global grupo
174
175
176     [a1, b1, c1, d1, e1] = ["", "", "", "", ""]
177
178     pregunta = Pregunta()
179     if grupo == "":
180         [a1, b1, c1, d1, e1] = genera_frase(request)
181     elif grupo == "grupo1":
182         [a1, b1, c1, d1, e1] = genera_frase_g1(request)
183     elif grupo == "grupo2":
184         [a1, b1, c1, d1, e1] = genera_frase_g2(request)
185     elif grupo == "grupo3":
186         [a1, b1, c1, d1, e1] = genera_frase_g3(request)
```

*Ilustración 26: Extracto del código de generación de frases en función del Grupo de Aprendizaje*

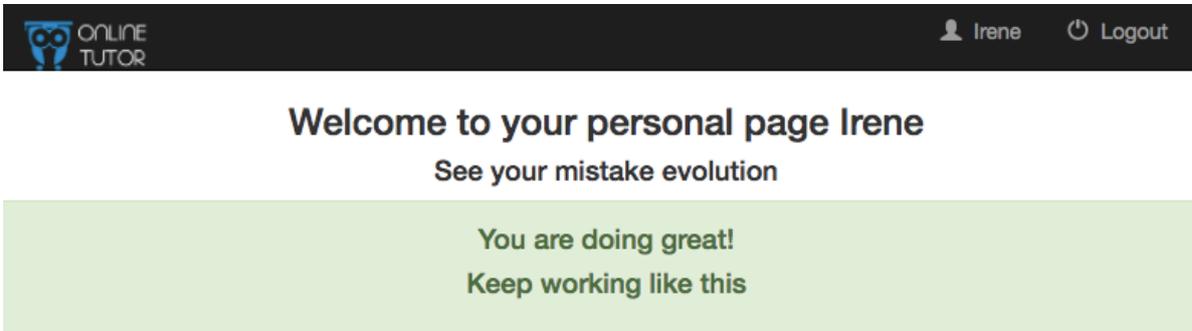
Tal y como se puede observar en la Ilustración 26, dependiendo del grupo que se haya asignado en la ronda anterior, se va a gestionar desde una función diferente. En el caso en el

cual no se haya asignado grupo anteriormente, por ejemplo, al acceder a la aplicación, se generarán preguntas teniendo en cuenta todos los conceptos estudiados.

#### 5.4.4 ANÁLISIS DE LA EVOLUCIÓN DE LOS ALUMNOS

Se ha considerado útil que el alumno sea capaz de observar cuál ha sido su evolución en el tiempo en cuanto al número de fallos que ha cometido. Por ese motivo, se ha habilitado una página personal en la cual el alumno puede observar un gráfico con su evolución.

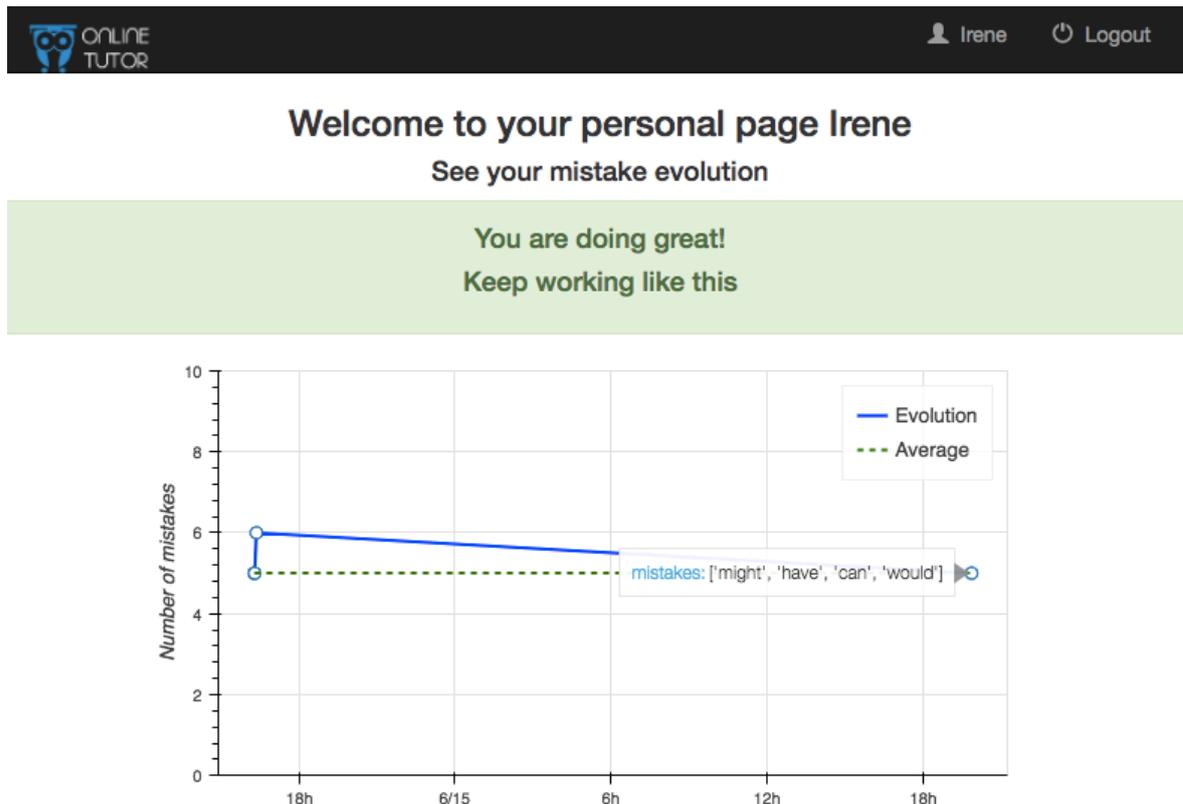
Además de la evolución del alumno, se puede ver cuál es la media de la clase. De esta manera se busca motivar al alumno, para que mejore sus resultados y pueda estar en la media o por encima de ella. También se incluyen frases para motivar al alumno en función de su evolución.



The screenshot shows the 'ONLINE TUTOR' interface for user 'Irene'. It includes a 'Logout' button and a personal page header: 'Welcome to your personal page Irene' and 'See your mistake evolution'. A green box contains the message: 'You are doing great! Keep working like this'. Below this is a line graph with 'Number of mistakes' on the y-axis (0 to 10) and time on the x-axis (18h, 6/15, 6h, 12h, 18h). The graph shows two lines: a solid blue line for 'Evolution' and a dashed green line for 'Average'. The 'Evolution' line starts at 6 mistakes at 18h, drops to 5 at 6/15, and then gradually decreases to 5 at 18h. The 'Average' line is constant at 5 mistakes.

*Ilustración 27: Ejemplo de evolución de un alumno*

Se ha añadido la posibilidad de poder ver cuáles han sido los fallos en cada uno de los intentos que el alumno ha realizado. Esta funcionalidad se ha pensado para que el alumno entienda cuales han sido los fallos concretos.



*Ilustración 28: Evolución del alumno y muestra de los fallos cometidos*

Para poder ver estos fallos simplemente se debe pasar el ratón por encima de cada uno de los círculos que simbolizan los intentos del alumno.

Para poder acceder a este gráfico, se puede pulsar el botón con el nombre del alumno en la barra de navegación. Este botón es accesible en cualquier momento de la navegación del alumno por la aplicación.

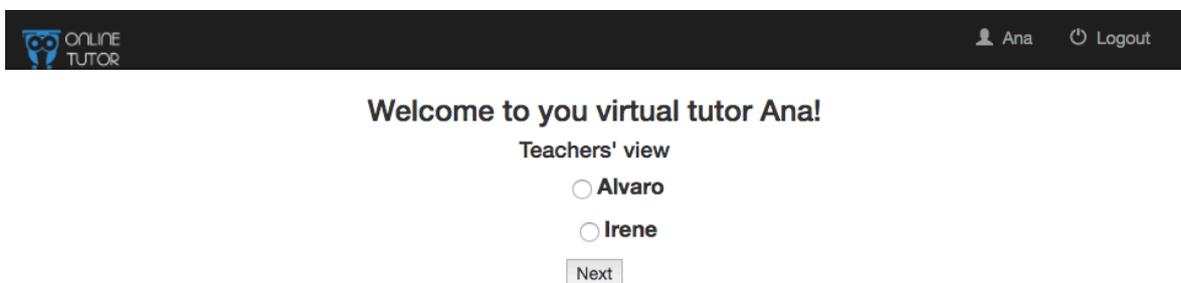
Este gráfico se ha realizado con la ayuda de una librería específica de Python llamada Bokeh, accesible desde <http://bokeh.pydata.org/>. Esta librería permite presentar gráficos elegantes de manera sencilla. Se trata de gráficos interactivos en los que el usuario de la aplicación puede acercarse, alejar o guardar la imagen.

Se ha elegido esta librería, ya que se podrían utilizar muchas otras, porque es una de las que mejor se adaptan a aplicaciones web [26].

### 5.4.5 VISTA DE PROFESOR

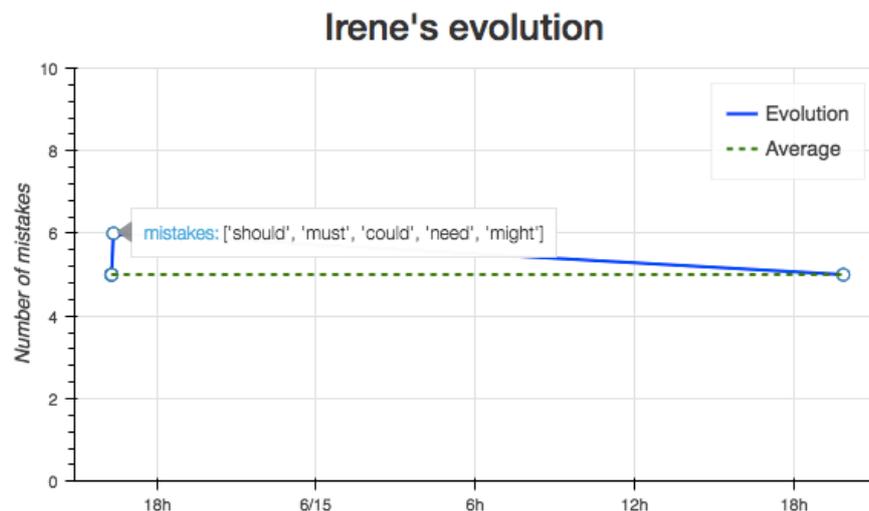
Dado que esta aplicación pretende ser un apoyo para alumno y profesor, se ha considerado un punto importante que el profesor encargado de una clase pueda acceder a la aplicación para observar el nivel de sus alumnos y cuál es su evolución. De esta manera el profesor sabrá cuáles son los principales problemas en general y cuáles son los problemas particulares de cada alumno. Así se podrá centrar en los alumnos con más problemas y podrá crear tutorías o clases de refuerzo.

En esta vista de profesor se ofrecerá una lista con los alumnos que pertenecen a una clase y se podrá elegir cada uno de los nombres para acceder a la evolución del alumno.



*Ilustración 29: Índice de la aplicación para un profesor*

Esta evolución se presenta de la misma manera que en el perfil personal de los alumnos. Además de la evolución del alumno, también se ofrece una comparación con la media de la clase. De este modo el profesor puede conocer cómo se sitúa el alumno con respecto al conjunto de la clase. También se ofrece la posibilidad de observar cuáles han sido los fallos de cada alumno en cada uno de los intentos, de la misma manera que en el perfil personal de cada alumno.



*Ilustración 30: Evolución de un alumno y comparación con la media de la clase*

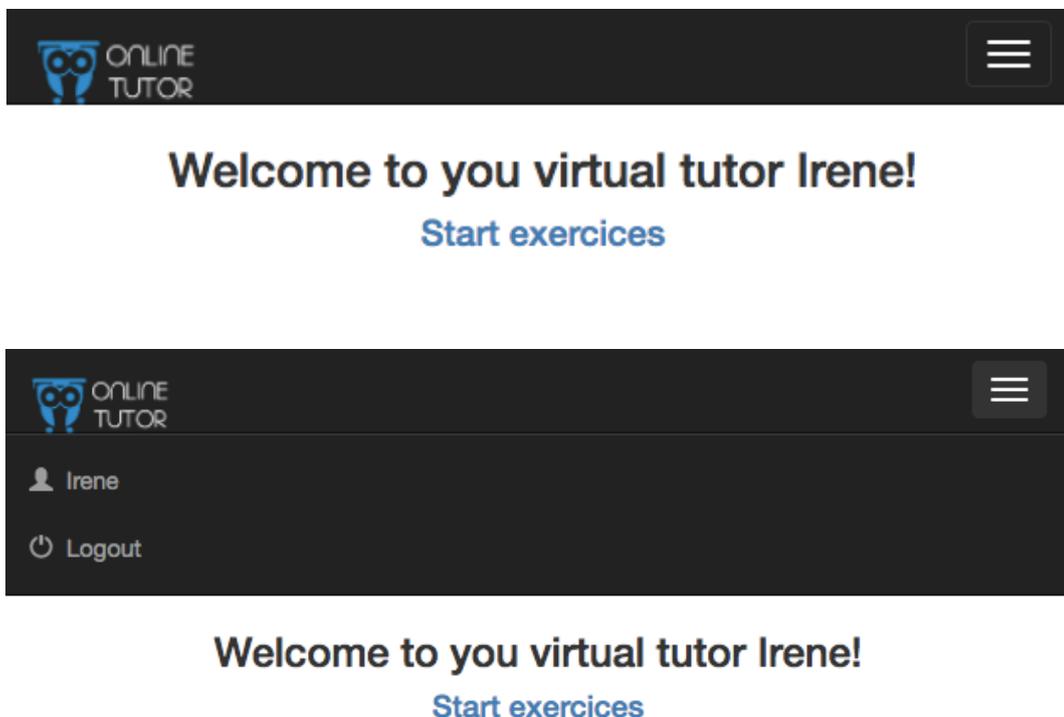
Para poder acceder a la vista de profesor, simplemente se debe introducir el usuario y contraseña de un usuario que haya sido catalogado en el grupo de los profesores.

## 5.5 DISEÑO GRÁFICO DE LA APLICACIÓN

Para el diseño gráfico de la aplicación se ha utilizado Bootstrap. Como se ha explicado anteriormente, esta herramienta es muy útil para dar formato a las aplicaciones web y que tengan una apariencia más atractiva. Se ha utilizado Bootstrap a partir de CDN, es decir a

través de scripts que se ejecutan cuando se carga el template HTML de una vista de la aplicación.

Gracias a Bootstrap se ha podido construir una barra de navegación y se han podido añadir iconos que hacen que la aplicación sea más *user friendly*. Además, Bootstrap permite que las páginas web sean responsivas y se pueda acceder a ellas desde cualquier dispositivo sin perjudicar el diseño.



*Ilustración 31: Vista de índice de la aplicación desde un dispositivo pequeño*

En la Ilustración 31 se puede observar cómo la barra de navegación ha cambiado con respecto a otras imágenes que se han visto de la aplicación. Esto se debe a la responsividad, ya que al detectar que el navegador ha reducido su tamaño la aplicación es capaz de adaptarse.

De manera a dar una imagen más natural a la aplicación se ha diseñado un logo. Este logo aparece a la izquierda de la barra de navegación y se puede pulsar en cualquier momento para acceder a la vista de inicio de la aplicación.



*Ilustración 32: Logo de la aplicación*

A la hora de diseñar el logo también se ha definido un nombre para la aplicación, *Online Tutor*.

Tanto el diseño gráfico de la web, cómo la creación de un logo representativo de la aplicación, hacen que esta tenga un aspecto moderno y actual, capaz de atraer a usuarios por su facilidad y comprensión de uso.

## Capítulo 6. COMPROBACIÓN DEL SISTEMA

### 6.1 CASOS EJEMPLO

En este apartado se va a mostrar el proceso que sigue tanto un alumno como un profesor durante el uso de la aplicación. Anteriormente, se han mostrado ejemplos y algunos de los resultados del desarrollo. Sin embargo, este apartado está pensado para constatar la línea del proceso que sigue cada uno de los usuarios claves del sistema.

Empezaremos por lo más básico, el sistema de autenticación.

**Login**

Username	<input type="text" value="Username"/>
Password	<input type="password" value="Password"/>
<input type="submit" value="Submit"/>	

*Ilustración 33: Vista del sistema de autenticación*

Esta es la primera vista que se tiene del sistema, ya que no se permite que un usuario cree una cuenta por sus propios medios. En el caso de los alumnos, su profesor será el encargado de crear su perfil. Para ello se ha proporcionado al usuario categorizado como Profesor la autoridad de crear perfiles y editarlos dentro de la interfaz de administrador de Django. La creación del perfil de los profesores se deberá gestionar de manera interna dentro de la aplicación, de esta manera se podrá verificar la identidad del mismo.

### 6.1.1 CASO DE UN ALUMNO

En este primer caso ejemplo, se seguirá el proceso de un alumno que accede a la aplicación por la primera vez. En este caso, la primera batería de preguntas se realizará de manera completamente aleatoria.



Ilustración 34: Vista de índice para un alumno

Para acceder a las preguntas, el alumno deberá simplemente pulsar el botón *Start exercises*. A partir de ese momento se empezarán los ejercicios, que se presentan tal y como se muestra en la Ilustración 35.

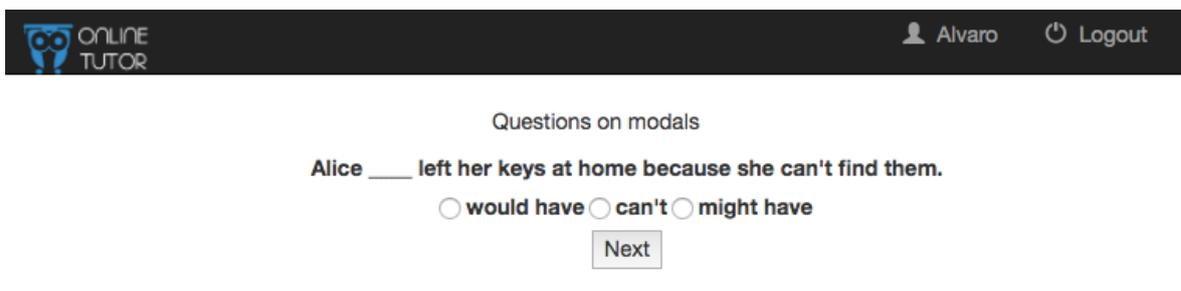


Ilustración 35: Vista de las preguntas propuestas a un alumno

Se propondrán al alumno diez preguntas como estas, y una vez que sean contestadas se ofrecerán los resultados de dichas preguntas. Hasta tres veces puede realizar el alumno estos ejercicios, tras las cuales se propondrá visitar al profesor para obtener una ayuda más precisa.



You got 7 errors!  
[See your answers](#)  
[Try again!](#)  
[Review concepts](#)

Ilustración 36: Resultados del ejercicio de un alumno

Una vez llegado a esta vista, el alumno puede ver cuáles han sido las preguntas que ha fallado. Para ello sólo debe pulsar sobre la opción *See your answers*.



### See the answers

Alice \_\_\_ left her keys at home because she can't find them.  
would have  
can't  
might have

You \_\_\_ like to go to the match with Sarah.  
might have  
would  
shouldn't have

Sara \_\_\_ like to go to the match with Sarah.  
would  
didn't have  
would

Taylor \_\_\_ forgotten about her last boyfriend. She is still very sad.  
may have  
may not have  
can

Ilustración 37: Extracto de las preguntas acertadas y falladas del alumno

Si pulsa sobre la opción *Review concepts*, el alumno podrá aprender más sobre los conceptos que se han identificado como problemáticos para el alumno.

The screenshot displays the 'Review the topics' section of the ONLINE TUTOR system. The interface is organized into a grid of six cards, each representing a different modal verb. Each card contains the verb name, a short explanation of its use, and a 'Read More' button. Below the grid is a 'Try again!' button. The top navigation bar includes the ONLINE TUTOR logo, the user's name 'Alvaro', and a 'Logout' button.

**Can**  
We use *can* to make statements of what is...  
[Read More](#)

**Have to**  
*Have to* is used to express strong obligation...  
[Read More](#)

**May**  
We use *may* when we are not sure about something...  
[Read More](#)

**Might**  
We use *might* when we are not sure about something...  
[Read More](#)

**Should**  
*Should* is used to express a mild obligation...  
[Read More](#)

**Would**  
We use *would* as the past tense of *will*...  
[Read More](#)

[Try again !](#)

Ilustración 38: Vista de revisión de conceptos

Pulsando la opción *Try again!*, el alumno puede volver a hacer los ejercicios.

Como se ha podido observar, en la barra de navegación aparece un icono de usuario al lado del nombre del alumno. En este caso el usuario de prueba se llama Álvaro. Pulsando en este icono el alumno puede acceder a su página personal en la cual podrá ver cuál ha sido su evolución.

Álvaro ya ha intentado hacer los ejercicios tres veces y no se le ha dado especialmente bien, lo podemos ver en la Ilustración 39.



## Welcome to your personal page Alvaro

See your mistake evolution

It looks like you are having a bad day...  
Don't give up yet !

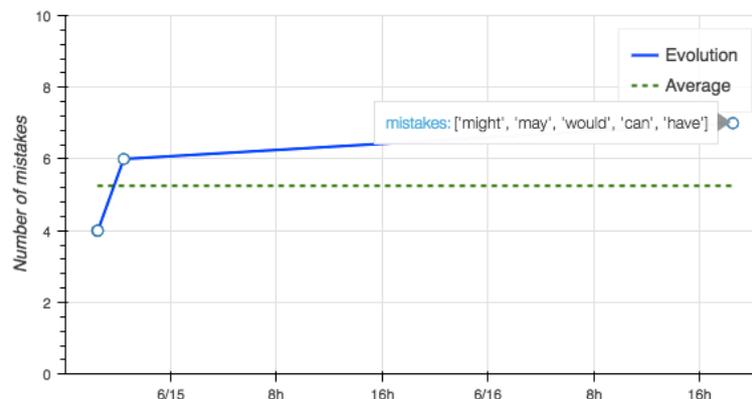


Ilustración 39: Vista del perfil personal del alumno

Como a Álvaro no se le han dado muy bien estos ejercicios –el número de fallos está en general por encima de la media de la clase-, se le va a intentar motivar para que no abandone y siga estudiando y practicando estos ejercicios. Además, Álvaro puede ver cuáles han sido sus fallos en cada uno de los intentos que ha realizado.

### 6.1.2 CASO DE UN PROFESOR

En el caso de los profesores, al introducir su nombre de usuario y contraseña pueden acceder a una lista de los alumnos que pertenecen a su clase. El profesor puede elegir cualquiera de estos alumnos para acceder a su evolución.



### Welcome to you virtual tutor Ana!

Teachers' view

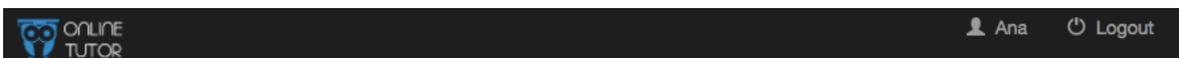
Alvaro

Irene

Next

Ilustración 40: Vista de un profesor

Ana es una profesora de inglés, y en su clase tiene a dos alumnos: Álvaro e Irene. Ana está preocupada por la evolución de Álvaro ya que le ha notado un poco flojo en los últimos exámenes realizados. Ana puede seguir la evolución de Álvaro seleccionando su nombre y pulsando sobre el botón *Next*.



### Alvaro's evolution

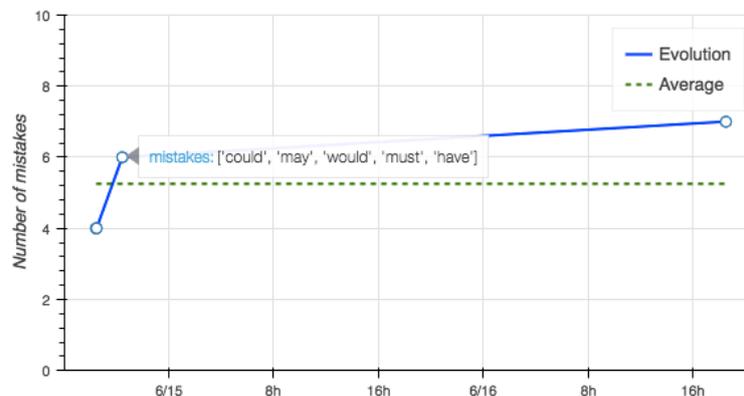


Ilustración 41: Evolución de un alumno, vista de profesor

Con esta herramienta, la profesora puede ver cuál ha sido la evolución de los ejercicios realizados por Álvaro y además puede ver cuáles han sido los fallos que ha cometido en cada uno de los casos. De esta manera la profesora podrá conocer cuáles son los conceptos que Álvaro no entiende y podrá dar un enfoque adaptado a los mismos.

### **6.1.3 CASO DE UN NUEVO DESARROLLADOR**

Para un desarrollador que quiera crear un proyecto similar, es necesario haber entendido el proceso que se ha descrito a lo largo de este documento. Los pasos a seguir para poder construir una plataforma similar serían los siguientes, basado en los pasos que la autora de esta tesis ha seguido:

- Creación de un conjunto de entrenamiento: un buen conjunto de datos es esencial para poder realizar un sistema experto de cualquier tipo. Si no se cuenta con un experto al cual se le puedan realizar entrevistas para entender los conceptos principales de la materia, este conjunto de entrenamiento es el que se va a utilizar para obtener las reglas de inferencia en las que se base el sistema.
- Análisis: Esta tesis de máster se ha apoyado mucho en la observación y el análisis de los datos. Es necesario realizar un análisis adecuado a los objetivos que se quieren conseguir con el sistema que se está desarrollando.
- Programación: Siempre es más sencillo utilizar un lenguaje de programación que haya utilizado anteriormente y con el que uno se sienta cómodo. Si no se ha dado el caso, como para este proyecto, lo mejor es buscar un lenguaje de programación que se adapte a las necesidades del sistema que se quiere desarrollar. En este caso, se ha elegido programar con Python dado que cuenta con una API muy completa para el análisis de datos.

## **6.2 DIFICULTADES ENCONTRADAS**

Esta tesis de máster ha estado compuesta por muchos retos y dificultades que se han ido resolviendo. El primer reto encontrado fue el de encontrar datos útiles para crear el conjunto de entrenamiento. El conjunto de entrenamiento ha sido el componente principal de este trabajo y era necesario que los datos utilizados fueran suficientes tanto en calidad como en

cantidad. Gracias a la ayuda del experto en el que se ha confiado, se pudo acceder a unos datos con los que se pudo crear el conjunto de entrenamiento.

Uno de los retos más importantes a los que la autora de la tesis se ha enfrentado, es a la programación de la aplicación web. Python era un lenguaje desconocido para ella, y ha tenido que pasar bastante tiempo realizando tutoriales y buscando información para poder empezar a programar. Además, también se ha debido trabajar por primera vez con Bootstrap y Django, lo que al principio resultó complicado. Sin embargo, se han convertido en herramientas muy útiles tanto para este proyecto como para futuros proyectos.

Globalmente, se han ido encontrando pequeñas dificultades a medida que se ha ido avanzando en el desarrollo de la tesis. Estas se han ido resolviendo, muchas veces con ayuda de internet, dado que algunos de los problemas ya se habían tratado por usuarios de internet en páginas como Stack Overflow [27].

## **Capítulo 7. CONCLUSIONES Y TRABAJOS FUTUROS**

A lo largo de este documento se han ido presentando el diseño, desarrollo y resultados finales de la tesis de máster. Tal y como se ha explicado al principio del documento, este proyecto se ha desarrollado con la idea de proporcionar un soporte para alumnos y profesores.

Tanto el objetivo principal como los secundarios, presentados en el Capítulo 4. se han cumplido. Se ha desarrollado una plataforma web capaz de generar ejercicios de manera aleatoria y adaptada a cada alumno. Esta plataforma realiza un análisis de las respuestas dadas por el alumno para obtener información sobre los errores que se han cometido. Al conocer los fallos que un alumno ha tenido, se le puede ofrecer información para que estudie los conceptos se han detectado como no entendidos y además se le pueden mostrar las preguntas que ha acertado y cuáles de ellas ha fallado.

Uno de los puntos más interesantes del sistema, es la posibilidad de seguir la evolución del alumno y compararla a la media de la clase. De esta manera se puede saber si el alumno va con retraso respecto al resto de los alumnos y se puede tratar antes de que el problema vaya a más. Además, pasando el ratón por encima de cada punto de la gráfica se puede obtener información sobre que fallos ha cometido concretamente el alumno en cada uno de los intentos que ha realizado. Se ha añadido una funcionalidad que consiste en mostrar alumno frases que le motiven y le ayuden a seguir practicando y aprendiendo.

También se ha creado una interfaz para que los profesores puedan acceder a la aplicación. De esta manera podrán observar la evolución de cada uno de los alumnos de su clase y tomar las medidas que ellos consideren necesarias.

Como trabajo futuro, se podría ampliar la funcionalidad de la aplicación. Con esto se quiere decir, ampliar la temática del soporte de aprendizaje. Esta tesis de máster se ha centrado en

el aprendizaje de los verbos modales en inglés, pero si se ampliara la base de datos se podría ofrecer un soporte para el aprendizaje de más aspectos del aprendizaje del inglés.

Igualmente, también se podría ampliar el espectro a otras materias ya que el modelo de la aplicación es muy flexible y permite la realización de bases de datos similares que se pueden utilizar con el algoritmo ya desarrollado para esta plataforma web.

## Capítulo 8. BIBLIOGRAFÍA

- [1] M. Á. Sanz Bobi, «Introducción a la Inteligencia Empresarial,» ICAI - Universidad Pontificia de Comillas, Madrid, 2016.
- [2] Y. Wang, «Application of Budget System Based on Business Intelligence,» *IEEE*, 2016.
- [3] A. P. Gálvez, *Business Intelligence y las tecnologías de la información*, 2ª edición ed., IT Campus Academy, 2016.
- [4] Logicalis, «Logicalis,» 2015. [En línea]. Available: <https://blog.es.logicalis.com/analytics/el-impacto-de-la-calidad-de-datos-en-entornos-business-intelligence>. [Último acceso: Abril 2017].
- [5] M. Á. García Soto, «Futuros administradores informáticos,» 2013. [En línea]. Available: <http://asir2-luisvives.blogspot.com.es/2013/01/fiabilidad-confidencialidad-integridad.html>. [Último acceso: Abril 2017].
- [6] Cenatic (Ministerio de Energía, Turismo y Agenda Digital), «Cenatic,» 2013. [En línea]. Available: <http://www.cenatic.es/hemeroteca-de-cenatic/3-sobre-el-sector-del-sfa/40430-big-data-business-intelligence-data-mining-y-open-data-con-herramientas-de-codigo-abierto-y-sin-coste-por-primera-vez-juntos-en-la-jornada-open-analytics>. [Último acceso: Abril 2017].
- [7] Matlab, «Matlab,» 2017. [En línea]. Available: <https://es.mathworks.com/products/matlab.html>. [Último acceso: Abril 2017].

- [8] Weka, «Weka, The University of Waikato,» 2017. [En línea]. Available: <http://www.cs.waikato.ac.nz/ml/weka/>. [Último acceso: Mayo 2017].
- [9] M. Á. Sanz Bobi, *Breve introducción a WEKA*, 2016.
- [10] Wikipedia, «Wikipedia,» [En línea]. Available: <https://es.wikipedia.org/wiki/Python>. [Último acceso: Junio 2017].
- [11] Django, «Django,» [En línea]. Available: <https://www.djangoproject.com/>. [Último acceso: Junio 2017].
- [12] Django, «Django,» [En línea]. Available: <https://www.djangoproject.com/start/>. [Último acceso: 2 2017].
- [13] T. N. Boston, «Youtube,» 2016. [En línea]. Available: <https://www.youtube.com/watch?v=qgGIqRFvFFk&list=PL6gx4Cwl9DGBImzzFcLgDhKTTfNlfX1IK&index=1>. [Último acceso: Febrero 2017].
- [14] JetBrains, «Pycharm,» [En línea]. Available: <https://www.jetbrains.com/pycharm/>. [Último acceso: Junio 2017].
- [15] Bootstrap, «Bootstrap,» [En línea]. Available: <http://getbootstrap.com/>. [Último acceso: Mayo 2017].
- [16] A. Azevedo y M. Filipe Santos, «Business Intelligence: State of the Art, Trends and Open Issues,» International Conference on Knowledge Management and Information Sharing, 2009.
- [17] B. Kurshan, «The Future of Artificial Intelligence in Education,» *Forbes*, 2016.
- [18] W. Ma y O. O. Adesope, «Intelligent Tutoring Systems and Learning Outcomes: A Meta-Analysis,» *Journal of Educational Psychology*, vol. 106, nº 4, pp. 901-918, 2014.

- [19] L. Kurup, A. Joshi y N. Shekhokar, «Intelligent Tutoring System for learning English Punctuation,» de *2016 International Conference on Computing Communication Control and Automation (ICCUBEA)*, Pune, India, 2016.
- [20] IBM, «IBM Watson,» [En línea]. Available: <https://www.ibm.com/watson/education/>. [Último acceso: Junio 2017].
- [21] B. Dickson, «How Artificial Intelligence enhances education,» *The Next Web*, 13 Marzo 2017. [En línea]. Available: [https://thenextweb.com/artificial-intelligence/2017/03/13/how-artificial-intelligence-enhances-education/#.tnw\\_P4vAUdWk](https://thenextweb.com/artificial-intelligence/2017/03/13/how-artificial-intelligence-enhances-education/#.tnw_P4vAUdWk). [Último acceso: Junio 2017].
- [22] Y. M. Tashtoush, M. Al-Soud y M. Fraihat, «Adaptive e-learning web-based English tutor using data mining techniques and Jackson's learning styles,» de *8th International Conference on Information and Communication Systems*, Irbid, Jordan, 2017.
- [23] E. Grant y K. Payne, *View Points for Bachillerato 1*, Burlington Books.
- [24] E. Grant y K. Payne, *View Points for Bachillerato 2*, Burlington Books.
- [25] British Council, «British Council LearnEnglish,» [En línea]. Available: <https://learnenglish.britishcouncil.org/fr/english-grammar/verbs/modal-verbs/can-could-and-could-have>. [Último acceso: Junio 2017].
- [26] Flowing Data, «FlowingData,» 2013. [En línea]. Available: <https://flowingdata.com/2013/11/22/bokeh-a-python-library-for-interactive-visualization/>. [Último acceso: Mayo 2017].
- [27] Stack Overflow, «Stack Overflow,» [En línea]. Available: <https://es.stackoverflow.com/>. [Último acceso: 2017].

## ANEXO A – CÓDIGOS DE RESPUESTA

Se recuerda que las respuestas dadas por los alumnos han sido tratadas y codificadas para su análisis. En la siguiente tabla se encuentran los códigos que se han utilizado.

Código	Significado
0	correcto
1	can
2	can't
3	could
4	couldn't
5	have
6	don't have
7	may
8	may not
9	might
10	might not
11	must
12	mustn't
13	need
14	don't need
15	shall
16	shall not
17	should
18	shouldn't
19	would
20	wouldn't
21	can have
22	can't have
23	could have
24	couldn't have
25	had
26	didn't have
27	may have
28	may not have
29	might have
30	might not have
31	must have
32	mustn't have
33	needed
34	didn't need
35	should have
36	shouldn't have
37	would have
38	wouldn't have

<b>Código</b>	<b>Significado</b>
<b>39</b>	NA
<b>40</b>	significado
<b>41</b>	no modal

Estos códigos sirven para identificar el tipo de error que el alumno ha cometido. Si se esperaba cómo respuesta *can* y se ha respondido *can't*, se pondrá el código de error 2.

El código 39 significa que el alumno no ha respondido la pregunta. El código 40 se refiere a las respuestas en las cuales se observa que el alumno no ha entendido el significado de la frase y no ha sabido responder correctamente. El código 41 significa que en su respuesta el alumno no ha utilizado ningún verbo modal.

## ANEXO B – GRUPOS DE APRENDIZAJE

Grupo 1	Grupo 2	Grupo 3	Grupo 4	Grupo 5	Grupo 6	Grupo 7	Grupo 8	Grupo 9
1	11	18	2	31	52	55	53	54
3	15	33	13	39	58	56	66	57
4	16	46	32	49	71	59	70	62
5	37	50			72	60		
6					73	61		
7						63		
8						64		
9						65		
10						67		
12						68		
14						69		
17						74		
19								
20								
21								
22								
23								
24								
25								
26								
27								
28								
29								
30								
34								
35								
36								
38								
40								
41								
42								
43								
44								
45								
47								
48								
51								

## ANEXO C – GRUPOS DE APRENDIZAJE Y RECOMENDACIONES

Grupos alumnos	Recomendaciones
<b>G1</b>	GA1, GA3, G4, G5, G8, GA7
<b>G2</b>	G1, G3, G4, G5, G8, G9
<b>G3</b>	G3, G4, G5
<b>G4</b>	G1, G3, G6
<b>G5</b>	G1, G2, G3, G4, G5, G6, G7, G8, G9
<b>G6</b>	G6, G7
<b>G7</b>	G3, G4, G7, G8, G9
<b>G8</b>	G3, G4, G5, G7
<b>G9</b>	G1, G2, G3, G4, G5, G6, G7, G8, G9

Los grupos de recomendación se utilizarán para asignar al tutor virtual que tipos de ejercicio se tienen que generar en cada caso. Cada grupo de aprendizaje consta de formas verbales que contienen el mismo modal, tanto en forma negativa como positiva y en forma pasada.

G1

Can, can't, can have, can't have

G2

Could, couldn't, could have, couldn't have

G3

Have, don't have, had, didn't have

G4

May, may not, may have, may not have

*ANEXO C – GRUPOS DE APRENDIZAJE Y RECOMENDACIONES*

---

G5

Might, might not, might have, might not have

G6

Must, mustn't, must have, mustn't have

G7

Need, don't need, needed, didn't need

G8

Should, shouldn't, should have, shouldn't have

G9

Would, wouldn't, would have, wouldn't have