



**UNIVERSIDAD DE CASTILLA-LA MANCHA  
ESCUELA SUPERIOR DE INFORMÁTICA**

**GRADO EN INGENIERÍA EN INFORMÁTICA**

**PedInf: Sistema Inteligente de Ayuda a la  
Decisión en Enfermedades Infecciosas Pediátricas**

**Sergio Sevilla Ballesteros**

**Julio, 2021**



**PEDINF: SISTEMA INTELIGENTE DE AYUDA A LA DECISIÓN EN  
ENFERMEDADES INFECCIOSAS PEDIÁTRICAS**





**UNIVERSIDAD DE CASTILLA-LA MANCHA  
ESCUELA SUPERIOR DE INFORMÁTICA**

**Tecnologías y Sistemas de Información**

**GRADO EN INGENIERÍA EN INFORMÁTICA  
INTENSIFICACIÓN DE COMPUTACIÓN**

**PedInf: Sistema Inteligente de Ayuda a la  
Decisión en Enfermedades Infecciosas Pediátricas**

**Autor: Sergio Sevilla Ballesteros**

**Tutor académico: José Ángel Olivas Varela**

**Julio, 2021**



## **Sergio Sevilla Ballesteros**

Ciudad Real – Spain

*E-mail:* Sergio.Sevilla@alu.uclm.es

*Teléfono:* 654 067 118

© 2021 Sergio Sevilla Ballesteros

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Se permite la copia, distribución y/o modificación de este documento bajo los términos de la Licencia de Documentación Libre GNU, versión 1.3 o cualquier versión posterior publicada por la *Free Software Foundation*; sin secciones invariantes. Una copia de esta licencia está incluida en el apéndice titulado «GNU Free Documentation License».

Muchos de los nombres usados por las compañías para diferenciar sus productos y servicios son reclamados como marcas registradas. Allí donde estos nombres aparezcan en este documento, y cuando el autor haya sido informado de esas marcas registradas, los nombres estarán escritos en mayúsculas o como nombres propios.



**TRIBUNAL:**

**Presidente:**

**Vocal:**

**Secretario:**

**FECHA DE DEFENSA:**

**CALIFICACIÓN:**

**PRESIDENTE**

**VOCAL**

**SECRETARIO**

Fdo.:

Fdo.:

Fdo.:



# Resumen

Las enfermedades infecciosas representan un problema real en la sociedad, como buena prueba de ello es la pandemia provocada por el virus SARS-CoV-2 a la que el planeta se está enfrentando en estos momentos. Sin embargo, antes de esta pandemia, diferentes enfermedades infecciosas endémicas, como son la gripe o la malaria, han acabado con la vida de muchas personas alrededor del mundo, y sobretodo, con la vida de muchas personas en edad pediátrica. Por suerte, en los últimos años, la medicina ha avanzado de forma veloz tanto en su propio campo como en campos externos, como la inteligencia artificial.

La inteligencia artificial, gracias a sus diferentes campos, permite el desarrollo de sistemas inteligentes de ayuda a la decisión en el ámbito sanitario. Haciendo uso de la ingeniería del conocimiento, se puede extraer conocimiento de un experto en la materia, para conceptualizar y representar dicho conocimiento mediante reglas de producción. La analítica de datos por su parte, permite extraer patrones de conjuntos de datos, haciendo uso del aprendizaje automático, gracias a sus técnicas de clasificación, regresión y clustering entre otros.

En este TFG se presenta un sistema de ayuda a la decisión en enfermedades infecciosas pediátricas, combinando el uso de estos dos campos de la inteligencia artificial. Por un lado, se hace uso de la metodología IDEAL, que muestra la secuencia de pasos a llevar a cabo para la construcción de un sistema experto haciendo uso de la ingeniería del conocimiento. Por el otro lado, el desarrollo del análisis de datos será ejecutado sobre diferentes conjuntos de datos almacenados en un data lake, siguiendo el proceso KDD, que ayuda con sus fases a la construcción de modelos válidos y útiles para extraer conocimiento a través de los datos.

De esta forma, se obtiene un sistema inteligente de ayuda a la decisión basado en un sistema experto alimentado con un total de 74 reglas, apoyado en un proceso de análisis de datos que corrobora el conocimiento extraído y lo amplia, gracias al uso de clustering y árboles de decisión. Este sistema inteligente además puede ser usado por todo el mundo tras haber sido desarrollada una interfaz para el mismo, para permitir que con el simple manejo de la misma, se pueda acceder al conocimiento extraído en su totalidad. Tras ello, el sistema en su completitud se sometió a un proceso de evaluación, a base de casos de prueba de diferentes diagnósticos propuestos por el experto, el cual avaló los resultados obtenidos por el sistema.



# Abstract

Infectious diseases represent a real problem in society, a good example is the pandemic caused by the SARS-CoV-2 virus that the world is currently facing. However, before this pandemic, different endemic infectious diseases, such as influenza or malaria, have killed many people around the world, and above all, many people of paediatric age. Fortunately, in recent years, medicine has advanced rapidly, both through advances in its own field and in external fields, such as artificial intelligence.

Artificial intelligence, thanks to its different fields, allows the development of intelligent decision support systems in the healthcare field. By making use of knowledge engineering, knowledge can be extracted from an expert in the field, to conceptualise and represent this knowledge by means of production rules. Data analytics, on the other hand, makes it possible to extract patterns from data sets, making use of machine learning, thanks to its classification, regression and clustering techniques, among others.

This final degree project presents a decision support system for paediatric infectious diseases, combining the use of these two fields of artificial intelligence. On the one hand, use is made of the IDEAL methodology, which shows the sequence of steps to be carried out for the construction of an expert system using knowledge engineering. On the other hand, the development of the data analysis will be executed on different data sets of a data lake, following the KDD process, which helps with its phases to build valid and useful models to extract knowledge through data.

In this way, an intelligent decision support system is obtained based on an expert system fed with a total of 74 production rules, supported by a data analysis process that corroborates the extracted knowledge and extends it, thanks to the use of clustering and decision trees. This intelligent system can also be used by everyone after an interface has been developed to allow access to the extracted knowledge in its entirety by simply using the interface. After that, the system was subjected to an evaluation process based on test cases of different diagnoses proposed by the expert, who endorsed the results obtained by the system.



# Agradecimientos

Echando la vista atrás al verano de 2017, era prácticamente inimaginable para aquel chaval, bastante indeciso con lo que podría llegar a ser su futuro, pensar todo lo que esta experiencia le iba a aportar en diferentes ámbitos de su vida. Obviamente, el camino que he vivido tanto en estos cuatro años como en los años previos, no lo he recorrido solo, y solo tengo palabras de agradecimiento para aquellos que han estado conmigo.

Gracias a mis padres, Jose María y María Isabel, por transmitirme todos los valores que me conforman a día de hoy, desde la humildad y la bondad hasta el trabajo continuo y perseverante para poder lograr todos los objetivos que me marque a lo largo de mi vida.

Gracias a mi hermano, Javier, que siempre ha sido una referencia y una persona donde intentar verme reflejado. Has sido en parte el culpable de que haya escogido este camino, que sin duda, ha sido el correcto.

Gracias a mis abuelos, Guadalupe, Antonio, Petra y Jose María; fuisteis, sois y seréis siempre un ejemplo para mí. Gracias tanto por darle ese cariño al niño que fui como por aportarme una visión de la vida más austera y tan necesaria.

Gracias a mis amigos, tanto a los de siempre como a los que he tenido la oportunidad de conocer a lo largo de estos cuatro años, por estar ahí en todo momento y haber hecho más amenos momentos donde no todo venía de cara.

Agradecer también a José Ángel, que desde su posición de tutor me ha facilitado el desarrollo de este proyecto, ofreciéndome todo aquel conocimiento que he necesitado a lo largo de estos meses, además de aportarme una serie de consejos que me han sido muy útiles tanto para este proyecto como para futuros que estén por venir.

Sin duda, esta experiencia de cuatro años ha sido increíble y maravillosa, y la llevaré siempre conmigo. Muchas gracias a todos.

Sergio



*“If people never did silly things,  
nothing intelligent would ever get done.”*  
– Ludwig Wittgenstein



# Índice general

<b>Resumen</b>	<b>V</b>
<b>Abstract</b>	<b>VII</b>
<b>Agradecimientos</b>	<b>IX</b>
<b>Índice general</b>	<b>XIII</b>
<b>Índice de cuadros</b>	<b>XVII</b>
<b>Índice de figuras</b>	<b>XIX</b>
<b>Índice de listados</b>	<b>XXI</b>
<b>Listado de acrónimos</b>	<b>XXIII</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	2
1.2. Estrategia . . . . .	3
1.3. Estructura del documento . . . . .	4
<b>2. Objetivos</b>	<b>5</b>
2.1. Objetivo general . . . . .	5
2.2. Objetivos específicos . . . . .	5
<b>3. Antecedentes: Inteligencia artificial aplicada a la medicina</b>	<b>7</b>
3.1. Informática médica . . . . .	7
3.2. Inteligencia artificial en el ámbito sanitario . . . . .	9
3.3. Ingeniería del conocimiento . . . . .	10
3.4. Sistema experto . . . . .	12
3.5. Data lake . . . . .	13
3.6. Knowledge Discovery in Databases . . . . .	14

## 0. ÍNDICE GENERAL

3.7. Web scraping . . . . .	17
3.8. Estudios anteriores . . . . .	18
<b>4. Metodologías y herramientas</b>	<b>21</b>
4.1. Metodologías ágiles . . . . .	21
4.2. Scrum . . . . .	23
4.3. Metodología IDEAL . . . . .	25
4.4. Proceso KDD . . . . .	28
4.5. Metodología de creación de interfaces centradas en el usuario . . . . .	30
4.6. Aplicación de las metodologías en el proyecto . . . . .	36
4.6.1. Sprint 1 . . . . .	36
4.6.2. Sprint 2 . . . . .	36
4.6.3. Sprint 3 . . . . .	37
4.7. Herramientas . . . . .	37
4.7.1. CLIPS . . . . .	37
4.7.2. Google Colab . . . . .	39
4.7.3. Pandas y Scikit-learn . . . . .	40
4.7.4. PyQt . . . . .	41
4.7.5. GitHub . . . . .	41
<b>5. Sistema Inteligente PedInf</b>	<b>43</b>
5.1. Sistema experto . . . . .	43
5.1.1. Propuesta . . . . .	43
5.1.2. Estudio de viabilidad . . . . .	44
5.1.3. Adquisición del conocimiento . . . . .	48
5.1.4. Conceptualización . . . . .	49
5.1.5. Representación del conocimiento . . . . .	50
5.2. Análisis de datos . . . . .	52
5.2.1. Presentación del data lake . . . . .	52
5.2.2. Selección de datos . . . . .	56
5.2.3. Preproceso y transformación de datos . . . . .	60
5.2.4. Minería de datos e interpretación del conocimiento . . . . .	62
5.3. Interfaz de usuario . . . . .	70
5.3.1. Estructuración, reconocimiento y exploración . . . . .	70
5.3.2. Modelado, ideación y prototipado . . . . .	70
5.3.3. Formalización e implementación . . . . .	70

5.3.4. Validación y despliegue . . . . .	71
<b>6. Evaluación</b>	<b>73</b>
6.1. Evaluación del sistema experto en CLIPS . . . . .	73
6.2. Evaluación del árbol de decisión obtenido . . . . .	74
6.3. Evaluación del sistema en su completitud . . . . .	75
<b>7. Conclusiones</b>	<b>77</b>
7.1. Objetivos alcanzados . . . . .	77
7.2. Futuras mejoras . . . . .	78
7.3. Competencias . . . . .	79
<b>Referencias</b>	<b>81</b>
<b>A. Entrevistas con el experto</b>	<b>85</b>
A.1. Primera entrevista . . . . .	85
A.2. Segunda entrevista . . . . .	88
A.3. Tercera entrevista . . . . .	91
A.4. Cuarta entrevista . . . . .	95
<b>B. Clusters obtenidos</b>	<b>97</b>
<b>C. Árbol de Decisión</b>	<b>99</b>
<b>D. Código del Sistema Experto</b>	<b>105</b>
D.1. Fragmento de código del sistema experto en CLIPS . . . . .	105
D.2. Fragmento de código del sistema experto en Clipsy . . . . .	106



# Índice de cuadros

4.1. Pila de sprint: Sprint 1 . . . . .	36
4.2. Pila de sprint: Sprint 2 . . . . .	36
4.3. Pila de sprint: Sprint 3 . . . . .	37
5.1. Test de Slagel: Plausibilidad . . . . .	45
5.2. Test de Slagel: Justificación . . . . .	45
5.3. Test de Slagel: Adecuación . . . . .	46
5.4. Test de Slagel: Éxito . . . . .	47
5.5. Tabla Objeto – Atributo – Valor: Paciente . . . . .	50
5.6. Tabla Objeto – Atributo – Valor: Enf. Infecciosa . . . . .	50
5.7. Tabla Objeto – Atributo – Valor: Tratamiento . . . . .	50
5.8. Selección de datos: Casos clínicos . . . . .	59
5.9. Selección de datos: OMS . . . . .	59
5.10. Preproceso de datos: Casos clínicos . . . . .	61
5.11. Preproceso de datos: OMS . . . . .	61
5.12. Minería de datos: OMS . . . . .	64



# Índice de figuras

1.1. Hospitales atendiendo pacientes enfermos en la pandemia de gripe de 1918 (izq.) y en la pandemia de COVID-19 (der.) . . . . .	2
1.2. Esquema de la combinación de ambos procesos. . . . .	3
3.1. Composición de la informática médica. . . . .	8
3.2. Fases de la adquisición del conocimiento. . . . .	11
3.3. Clasificación de los sistemas expertos. . . . .	12
3.4. Estructura de un sistema experto. . . . .	12
3.5. Esquema de un data lake. . . . .	13
3.6. Fases del proceso KDD. . . . .	14
3.7. Ejemplo de uso del algoritmo K - means. . . . .	15
3.8. Árbol de decisión de la librería scikit-learn. . . . .	17
3.9. Ejemplos de cáncer de mama detectados en mamografías. . . . .	18
4.1. Reuniones de un sprint en Scrum. . . . .	24
4.2. Ciclo de trabajo en Scrum. . . . .	25
4.3. Bases de la metodología propuesta. . . . .	31
4.4. Fases de la metodología de interfaces gráficas centradas en el usuario. . . . .	32
4.5. Ejemplo de ejecución de entorno de razonamiento en CLIPS. . . . .	38
4.6. Diagrama de flujo para seleccionar el algoritmo a usar de Scikit-learn. . . . .	40
4.7. Control de versiones del proyecto. . . . .	41
5.1. Mapa de conocimientos del sistema experto. . . . .	51
5.2. Resultado de aplicar el método del codo desde 1 a 10 clusters. . . . .	62
5.3. Cantidad de datos representados. . . . .	63
5.4. Clusters obtenidos. . . . .	63
5.5. Gráfico personal sanitario. . . . .	64
5.6. Gráfico vacunación. . . . .	65
5.7. Gráfico saneamiento e higiene. . . . .	65
5.8. Ejemplo de ejecución de validación cruzada. . . . .	67

## 0. ÍNDICE DE FIGURAS

5.9. Árbol de decisión obtenido. . . . .	68
5.10. Prototipo desarrollado y presentado al usuario objetivo. . . . .	71
5.11. Interfaz implementada. . . . .	72
5.12. Diagrama de clases. . . . .	72
6.1. Ejecución del caso de prueba 1. . . . .	73
6.2. Ejecución del caso de prueba 2. . . . .	73
6.3. Árbol del caso de prueba 3. . . . .	74
6.4. Árbol del caso de prueba 4. . . . .	74
6.5. Ejecución del caso de prueba 5. . . . .	75
6.6. Ejecución del caso de prueba 6. . . . .	76
B.1. Países clasificados por cluster. . . . .	98
C.1. Parte más izquierda del árbol. . . . .	99
C.2. Parte izquierda del árbol. . . . .	100
C.3. Parte centro izquierda del árbol. . . . .	101
C.4. Parte central del árbol. . . . .	102
C.5. Parte derecha del árbol. . . . .	103

# Índice de listados

4.1. Definición de un deftemplate, una regla y un hecho en CLIPS . . . . .	38
4.2. Definición y ejecución del entorno de razonamiento con Clipspy . . . . .	39
5.1. Ejemplo de deftemplates y regla de producción . . . . .	51
5.2. Algoritmo de web scraping usado . . . . .	57
5.3. Hiperparametrización . . . . .	67
5.4. Reglas de producción de diagnóstico de úlcera péptica. . . . .	69
D.1. Código del sistema experto en CLIPS . . . . .	105
D.2. Código del sistema experto en Clipspy . . . . .	106
D.3. Reglas del sistema experto en Clipspy . . . . .	107



# Listado de acrónimos

<b>OMS</b>	Organización Mundial de la Salud
<b>UCLM</b>	Universidad de Castilla - La Mancha
<b>KDD</b>	Knowledge Discovery in Databases
<b>SBC</b>	Sistema Basado en el Conocimiento
<b>UCD</b>	User Centered Design
<b>GUI</b>	Graphical User Interface
<b>CLIPS</b>	C Languague Integrated Production System
<b>NASA</b>	National Aeronautics and Space Administration
<b>XML</b>	eXtensible Markup Language
<b>CSV</b>	Comma Separated Values
<b>PCA</b>	Principal Component Analysis
<b>GPU</b>	Graphics Processing Unit
<b>VM</b>	Virtual Machine
<b>RAM</b>	Random Access Memory
<b>PIB</b>	Producto Interior Bruto
<b>PDF</b>	Portable Document Format
<b>TFG</b>	Trabajo Final de Grado
<b>UML</b>	Unified Modeling Language



# Capítulo 1

## Introducción

ÚLTIMAMENTE algunos campos de la informática están evolucionando y progresando muy rápido, lo que está cambiando la forma de llevar a cabo diversas tareas en múltiples ámbitos de la sociedad. Uno de estos campos es la inteligencia artificial, la cual permite a sistemas tecnológicos percibir su entorno y relacionarse con él, aportándole la habilidad de presentar algunas capacidades propias del ser humano como el razonamiento, aprendizaje o planificación. Una de las razones que han impulsado con fuerza el crecimiento del campo de la inteligencia artificial es el crecimiento incesante del volumen de datos a la que cualquier interesado en esta materia tiene acceso hoy en día.

Otro factor que ha influido en el crecimiento de la inteligencia artificial es la increíble mejora que los computadores han experimentado en los últimos años, sobre todo debido a su gran crecimiento en capacidad de memoria y a las potentes unidades de cálculo de las que disponen actualmente, lo que permite al computador llevar a cabo sus tareas en un tiempo mucho más rápido y eficiente. De esta forma, aprovechando las ventajas mencionadas, y haciendo uso de la gran cantidad de datos públicos disponibles en la actualidad, gracias a las diferentes técnicas que conforman el campo de la inteligencia artificial, se pueden extraer conclusiones y conocimiento en un tiempo que un ser humano no puede igualar.

Esta rapidez a la hora de obtener resultados ha llamado la atención en la actualidad de múltiples negocios de diferentes ámbitos, quienes hacen uso de este campo de la informática para obtener ventajas sobre sus competidores directos. Sin embargo, la inteligencia artificial va más allá de los datos, ya que cuenta con otros campos como la lógica difusa, las redes neuronales o la ingeniería del conocimiento, las cuales, junto con el resto de campos, comparten el mismo fin, aportar a un sistema la capacidad de interpretar el conocimiento, para aprender de él y poder emplearlo para lograr las tareas y metas planteadas.

En la actualidad, el ser humano convive día a día con la inteligencia artificial y hace uso de ella, en muchos casos, sin darse cuenta. Algunos ejemplos son la inteligencia artificial integrada en vehículos, las redes sociales o los asistentes de voz. Sin duda, el crecimiento de este campo de la informática está cambiando la humanidad, ya que su evolución está generando cambios que pueden llegar a modificar la propia estructura de la sociedad.

## 1. INTRODUCCIÓN

### 1.1 Motivación

Las enfermedades infecciosas son aquellas causadas por microorganismos patógenos, como pueden ser bacterias, virus o parásitos. El gran inconveniente que presentan estas enfermedades es su capacidad de transmisión entre humanos, lo que a lo largo de la historia, ha puesto en jaque a la sociedad en varias ocasiones. Las pandemias que han generado la transmisión de estas enfermedades han provocado algunas de las mayores tragedias de la historia de la humanidad, como la pandemia de peste negra, que alcanzó su punto álgido entre los años 1347 y 1353, y que acabó con la vida de la mitad de la población europea, equivalente a una cifra de entre 75 y 200 millones de personas.

Sin ir más lejos, en la actualidad el planeta está viviendo la peor crisis sanitaria del último siglo, provocada por la expansión del virus SARS-CoV-2, causante de la enfermedad COVID-19, cuyo único final positivo posible va de la mano de la ciencia, gracias al desarrollo de vacunas y medicamentos que ayuden a combatir la enfermedad.



Figura 1.1: Hospitales atendiendo pacientes enfermos en la pandemia de gripe de 1918 (izq.) y en la pandemia de COVID-19 (der.)

Por desgracia, muchas de estas enfermedades no son erradicadas y continúan activas permanentemente, lo que recibe el nombre de enfermedad endémica. Algunas enfermedades endémicas comunes son la gripe, el ébola o la malaria. Uno de los aspectos más preocupantes de estas enfermedades es la gran virulencia con la que golpean cada año a personas en edad pediátrica, es decir, desde los 0 hasta los 14 años aproximadamente.

La OMS, en un artículo publicado en la revista científica *The Lancet*, [BMB03] reveló que 10 millones de niños perdieron la vida entre los años 2000 y 2003, y entre las principales causas (8 % de los decesos totales), se encuentra la malaria, una enfermedad infecciosa provocada por la picadura de mosquitos.

En este proyecto se hará uso de diversas técnicas de la inteligencia artificial para estudiar las diferentes causas que favorecen la existencia y expansión de enfermedades de este tipo y el diagnóstico de las mismas, dando como resultado final un sistema inteligente de ayuda a la decisión en enfermedades infecciosas pediátricas.

## 1.2 Estrategia

Para llevar a cabo la construcción del sistema inteligente de ayuda a la decisión, se usarán dos técnicas propias de la inteligencia artificial: el análisis de datos y el desarrollo de un sistema experto, al cual se le aportará el conocimiento extraído de un experto en la materia, haciendo uso de la ingeniería del conocimiento.

La unión de ambos procesos llegará a la hora de recabar resultados. Por un lado, del análisis de datos se obtendrán patrones, que serán evaluados a su vez por el sistema experto. De esta forma, nos aseguramos de que el conocimiento del experto aportado al sistema se ha gestionado adecuadamente y coincide con el conocimiento extraído del análisis de datos.

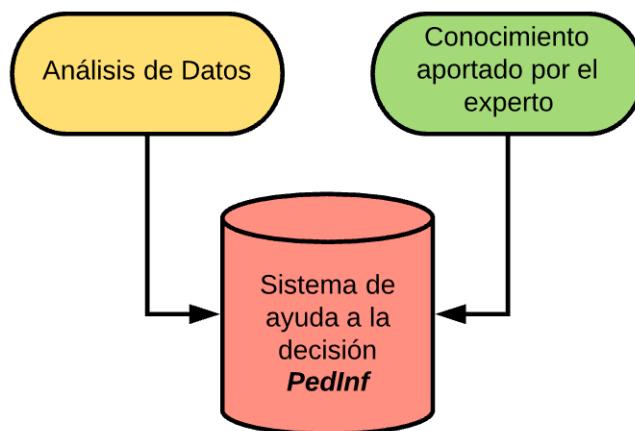


Figura 1.2: Esquema de la combinación de ambos procesos.

En el proceso de análisis de datos, además de trabajar en crear modelos que corroboren los diagnósticos del sistema experto, se llevará también a cabo un estudio de que factores afectan a la hora de la expansión de las enfermedades infecciosas en el planeta. Esta tarea se puede llevar a cabo gracias a los datos abiertos proporcionados por la OMS. Estos datos y los conceptos que se han nombrado a lo largo de este capítulo, serán presentados y desarrollados en próximos capítulos.

El experto que va a prestar su ayuda a lo largo del desarrollo del proyecto es Aitor Sánchez García, graduado en enfermería por la UCLM en el campus de Ciudad Real, que a lo largo de su grado, ha cursado diferentes materias que le han aportado conocimiento relacionado con la cuestión a tratar, además de haber realizado prácticas en centros médicos, donde ha tenido contacto con casos semejantes a los que se van a estudiar en el proyecto.

Para llevar a cabo la estrategia aquí definida a lo largo del proyecto, se hará uso de diferentes metodologías. El uso de las mismas permitirá que el desarrollo del proyecto se produzca de forma racional y eficiente, ayudando a alcanzar las metas y objetivos planteados de forma organizada.

## 1. INTRODUCCIÓN

### 1.3 Estructura del documento

En esta sección se incluye una descripción de los capítulos que se van a presentar al lector a lo largo de este documento, definiendo su contenido y finalidad, para facilitar la lectura del mismo.

#### **Capítulo 2: Objetivos**

En este capítulo se definen los objetivos a lograr gracias a la realización del proyecto, además de su finalidad y justificación.

#### **Capítulo 3: Antecedentes: Inteligencia artificial aplicada a la medicina**

En este capítulo se ofrece al lector una serie de conocimientos básicos para comprender el desarrollo del proyecto de forma adecuada, además de presentar estudios anteriores relacionados.

#### **Capítulo 4: Metodologías y herramientas**

Se presentan al lector la serie de metodologías usadas a lo largo del proyecto, así como sus distintas fases y etapas, y su aplicación. También se aporta al lector una breve descripción de las principales herramientas usadas a lo largo del proyecto.

#### **Capítulo 5: Sistema Inteligente PedInf**

Se exponen al lector los pasos llevados a cabo para el desarrollo del sistema de ayuda a la decisión generado en el proyecto, analizando por separado cada uno de los módulos principales del mismo.

#### **Capítulo 6: Evaluación**

Tras el desarrollo del sistema, se propondrá una serie de casos de prueba planteados por el experto, para comprobar si el comportamiento del mismo es el adecuado.

#### **Capítulo 7: Conclusiones**

Por último, se deberá de presentar cuales son las conclusiones del proyecto, explicando cuales de los objetivos han sido alcanzados de forma satisfactoria, así como posibles mejoras futuras.

Además de estos capítulos, se han creado los siguientes anexos, con el objetivo de ampliar la información ofrecida al lector a lo largo del documento.

#### **Anexo A: Entrevistas con el experto**

Entrevistas realizadas con el experto en la adquisición del conocimiento.

#### **Anexo B: Clusters obtenidos**

Clasificación de los diferentes países gracias a los clusters obtenidos.

#### **Anexo C: Árbol de Decisión**

Muestra el árbol de decisión en su completitud.

#### **Anexo D: Código del Sistema Experto**

Extracto del código del sistema experto desarrollado.

# Capítulo 2

## Objetivos

DURANTE este capítulo se pone en conocimiento del lector cuales son los objetivos a lograr para poder dar una solución al problema planteado. Los objetivos se desgranan a su vez en el objetivo general del proyecto y los objetivos específicos necesarios para obtener un resultado satisfactorio.

### 2.1 Objetivo general

El objetivo principal por el cual se implementa el sistema inteligente de ayuda a la decisión, es conseguir aplicar ciertos aspectos de la inteligencia artificial al ámbito sanitario. De esta forma, se busca contribuir y fomentar el uso de diferentes técnicas de la inteligencia artificial en áreas que no son comunes a ella, llevando a cabo las técnicas propias de la misma, para conseguir desarrollar, en este caso, un sistema inteligente de ayuda a la decisión en enfermedades infecciosas pediátricas.

Para conseguir crear dicho sistema de forma satisfactoria, se combinarán a lo largo del proyecto las dos técnicas nombradas anteriormente, el análisis de datos y la ingeniería del conocimiento, este último con el objetivo de desarrollar un sistema experto. A priori, ambos seguirán caminos diferentes en su desarrollo, pero siempre irán de la mano a la hora de trabajar con el experto, así como a la hora de producir sus resultados, que conformarán un sistema inteligente capaz de razonar de forma semejante a un pediatra a la hora de diagnosticar una enfermedad infecciosa.

### 2.2 Objetivos específicos

En los objetivos específicos se plasman las partes independientes del proyecto a desarrollar con valor para el lector por si mismas. Estos objetivos específicos muestran los principales entregables que el proyecto ofrece a los lectores.

#### Proceso de ingeniería del conocimiento

Conseguir modelar el conocimiento del experto mediante la realización de entrevistas, para poder llevar a cabo la extracción, conceptualización y representación del conocimiento. Gracias a dicho conocimiento, se podrá desarrollar el sistema experto planteado.

## 2. OBJETIVOS

### **Formación de un data lake**

Buscar datos relacionados con el asunto en cuestión tras charlar con el experto para, si es necesario debido a la gran cantidad de datos, su posterior almacenaje en un data lake que permita guardar datos tanto estructurados como no estructurados. El objetivo es poder llevar a cabo un proceso análisis de datos sobre aquellos conjuntos de datos que se desee, con el fin de extraer conocimiento a través de técnicas y algoritmos de machine learning.

### **Análisis de datos y machine learning**

Hacer uso de la analítica de datos y aplicación de algoritmos de machine learning sobre los conjuntos de datos del data lake que se consideren oportunos, con el objetivo de extraer patrones de conocimiento. Estos patrones mostrarán las tendencias que definen a los datos a la hora de cumplir una condición.

### **Validación del conocimiento extraído**

Corroborar que el conocimiento extraído tanto del análisis de datos como del sistema experto converge de forma satisfactoria, para poder así confirmar el correcto funcionamiento del sistema inteligente. La meta es conseguir un entorno de razonamiento correcto, completo y usable.

### **Desarrollo de una interfaz**

Diseñar una interfaz usable que permita a los usuarios objetivo manejar el sistema inteligente de forma cómoda y clara, para así, poner el conocimiento extraído al alcance de los usuarios.

## Capítulo 3

# Antecedentes: Inteligencia artificial aplicada a la medicina

**C**ON este capítulo se pretende aportar al lector los conocimientos necesarios para entender el desarrollo del proyecto correctamente. Para ello se va a describir la situación de la informática dentro de la sanidad a día de hoy, profundizando en el campo de la inteligencia artificial, para acabar concretando más en algunos términos esenciales para el desarrollo del proyecto.

### 3.1 Informática médica

El desarrollo de la medicina y de los sistemas sanitarios viene marcado tanto por el nivel de salud actual como por el nivel de desarrollo económico, científico y técnico. Dentro de dicho desarrollo científico, se encuentra el crecimiento de la informática en las últimas décadas. El primer uso de la informática en la salud se produjo en la década de 1950, cuando se empezaron a recoger datos dentales por parte de la Oficina Nacional de Normas de Estados Unidos. Desde entonces, la aplicación de la informática en el ámbito sanitario ha ido creciendo de forma apabullante. La consecuencia de tal crecimiento, fue la aparición del concepto informática médica.

La informática médica es la disciplina que aplica las ciencias de la información al contexto de la medicina. El principal objetivo es optimizar la adquisición, almacenamiento, recuperación y uso de la información en salud. [LFG13] Las herramientas propias de la informática médica incluyen computadores, software, sistemas de ayuda a decisiones clínicas, etc. Su ámbito se extiende desde aspectos clínicos hasta diversos aspectos de la atención de la salud, como enfermería, farmacia, salud pública e investigación médica.

La informática médica debe presentar un sistema de conocimientos y habilidades para ser capaz de mejorar las capacidades de los trabajadores de la salud. Para poder implementar la informática médica, es necesario el uso de software médico.

El software médico está compuesto por todos aquellos programas informáticos que son usados con finalidad médica. Algunos de sus usos son los siguientes:

- *Monitores.* Se hace uso de software especializado en interpretar la información de sensores, para mostrar el resultado en un monitor. A día de hoy se usan para medir la

### 3. ANTECEDENTES: INTELIGENCIA ARTIFICIAL APLICADA A LA MEDICINA

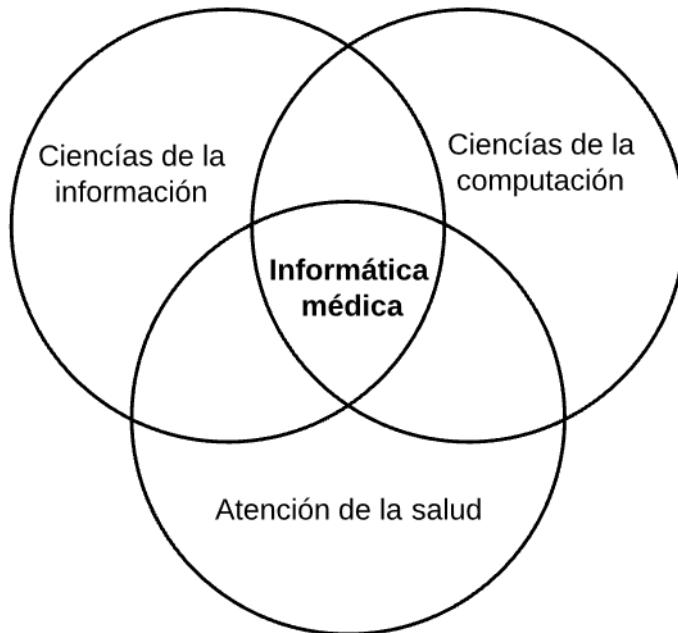


Figura 3.1: Composición de la informática médica.

frecuencia cardíaca, la presión arterial, tasa de respiración, etc.

- *Medicamentos bombas.* A día de hoy se usan dispositivos software que están programados para el bombeo de cierta cantidad de plasma, sangre o cualquier medicación en un paciente. El software tiene la capacidad de controlar muchos aspectos de los procedimientos del tratamiento.
- *Sistemas expertos.* Existen multitud de sistemas expertos para diferentes ámbitos sanitarios, como enfermería, medicina, farmacia, etc.
- *Terapia de entrega.* El software integrado en marcapasos y desfibriladores implantables proporciona tolerancia a fallos en tiempo real, críticos para la misión de vigilancia de los ritmos cardíacos y la entrega de terapia asociada.
- *Programas informáticos educativos.* Software utilizado como un centro de enseñanza o herramienta de estudio para los profesionales de la salud.
- *Software de gestión médica.* Herramienta que permite a través de la actualización de datos llevar a cabo la gestión de entidades dedicadas a la salud, tales como hospitales, sanatorios, clínicas, consultorios, etc.

Como se puede apreciar, el desarrollo de la informática ha mejorado la calidad de los servicios sanitarios. El uso de la informática en la salud es un fenómeno tan antiguo como los primeros sistemas informáticos, ya que con los mismos, se crearon sistemas de administración y procesamiento de datos de hospitales y clínicas.

La informática médica es esencial en países que puedan sufrir condiciones geográficas extremas, ya que gracias a la telemedicina, la distribución de imágenes y la evaluación remota

por expertos puede mejorar la calidad de vida de las personas que habiten dicha zona.

### 3.2 Inteligencia artificial en el ámbito sanitario

La inteligencia artificial cuenta con numerosas ramas, y cada una de ellas tiene importantes aplicaciones en el mundo sanitario. Aunque este campo de la informática es relativamente nuevo, tiene antecedentes de 1943, cuando el neurobiólogo Warren McCulloch y el lógico Walter Pitts publicaron un artículo donde trataban mediante la lógica proposicional los diferentes eventos neuronales y las relaciones existentes entre ellos. Lo que conllevó a que trece años después, en el año 1956, se celebrará la primera conferencia de inteligencia artificial en la universidad estadounidense de Dartmouth, en la cual se hicieron las primeras referencias a las redes neuronales artificiales. [ECA<sup>+03</sup>]

Desde aquel entonces la comunidad científica tuvo una percepción bastante optimista sobre lo que la inteligencia artificial podría ofrecer en diferentes ámbitos de la vida del ser humano, y entre ellos, el ámbito sanitario. Algunas aplicaciones de este campo informático tienen el objetivo de mejorar la atención al paciente, acelerando y consiguiendo mejores diagnósticos. Algunos ejemplos para diferentes áreas sanitarias son los siguientes:

- *Asistencial.* Existen algoritmos informáticos que ayudan a la prevención de enfermedades, como el cáncer de útero y de próstata. También existen programas informáticos que debido a su repetido uso son capaces tanto de crear diagnósticos acertados sobre diversas enfermedades como de recetar el tratamiento adecuado para paliar la enfermedad.
- *Investigación.* Muchas investigaciones médicas se ven ayudadas por el uso de la inteligencia artificial en ellas, ya que su utilización reduce los costes y facilita la obtención de datos. La lógica difusa está siendo aplicada en nuevas investigaciones, aportando nuevas visiones a la hora de crear modelos que estudien la expansión de enfermedades infecciosas y sistemas expertos de diagnóstico y tratamiento de enfermedades.
- *Gestión.* La gestión de centros sanitarios, en cuanto a pacientes, medios, herramientas y personal, se puede ver beneficiada del uso de la inteligencia artificial, mediante el uso de algoritmos de machine learning, que partiendo de registros históricos de datos, podrían establecer por ejemplo, cuál es la plantilla óptima para cada día de la semana dependiendo de las actividades y tareas a completar ese mismo día.

Ahora, una vez analizado como la inteligencia artificial es capaz de aportar conocimiento de forma útil al ámbito sanitario, se pasa a presentar al lector un conjunto de campos y conceptos relacionados con la inteligencia artificial que se usarán a lo largo del proyecto.

### 3.3 Ingeniería del conocimiento

La ingeniería del conocimiento es un campo de la inteligencia artificial que se dedica a estudiar, diseñar y desarrollar sistemas expertos, de los que se hablará a continuación. La ingeniería del conocimiento está centrada en el análisis y propuesta de métodos, con el objetivo de adquirir conocimiento humano, para guardar y representarlo, y posteriormente usarlo en procesos de razonamiento. Todo ello con el fin de emular en máquinas determinadas capacidades inteligentes propias del ser humano.

Por tanto, para obtener conocimiento sobre un tema específico, se debe determinar cual es el conocimiento que requiere un experto para hacerlo, tanto en el sentido teórico como en el experiencial. Al proceso de extraer dicho conocimiento del experto se le denomina adquisición del conocimiento, la cual suele requerir de mucho tiempo en el desarrollo de un sistema experto, entre otras causas, por la necesidad de encontrar el experto adecuado, la resistencia a abandonar secretos por parte del experto o por falta de disponibilidad del mismo en determinados momentos. Las etapas que debe de seguir un ingeniero de conocimiento para llevar a cabo una adquisición del conocimiento correcta, son los siguientes: [MP88]

- *Identificación del dominio y definición del problema:* se basa en identificar el alcance que tendrá el sistema y los problemas que podrán ser resueltos con su uso.
- *Identificación de conceptos y relaciones:* tanto el ingeniero del conocimiento como el experto deben de trabajar juntos con el fin de identificar conceptos. Los problemas que pueden surgir son la pérdida de conocimiento o la poca claridad de los conceptos
- *Formalización del conocimiento:* una vez conocidos los conceptos, se deben de formalizar, para poder introducirlos en la base de conocimientos, en la cual queden claras las relaciones entre ellos.
- *Chequeo:* se realizarán diferentes comprobaciones para evaluar la calidad y eficacia del conocimiento formalizado para descubrir posibles errores.
- *Refinado y validación:* una vez el sistema este desarrollado, se debe de considerar si se ha alcanzado la madurez suficiente. Para validar el sistema se le debe de poner a prueba, para comparar sus respuestas con las del experto para los mismos problemas.

Una de las técnicas más usadas para poner en marcha la adquisición del conocimiento, es realizar una serie de entrevistas al experto. Hay dos tipos de entrevistas, estructuradas y no estructuradas.

En una entrevista no estructurada el contenido de dicha entrevista no está predefinido. Sin embargo, el ingeniero del conocimiento requiere de cierta habilidad para que aunque la conversación no tenga guión, no acabe desembocando en una charla que no aporte nada al proceso de adquisición del conocimiento. Para ello, se suele fijar un tema generalizado, del cual, no se debe de salir.

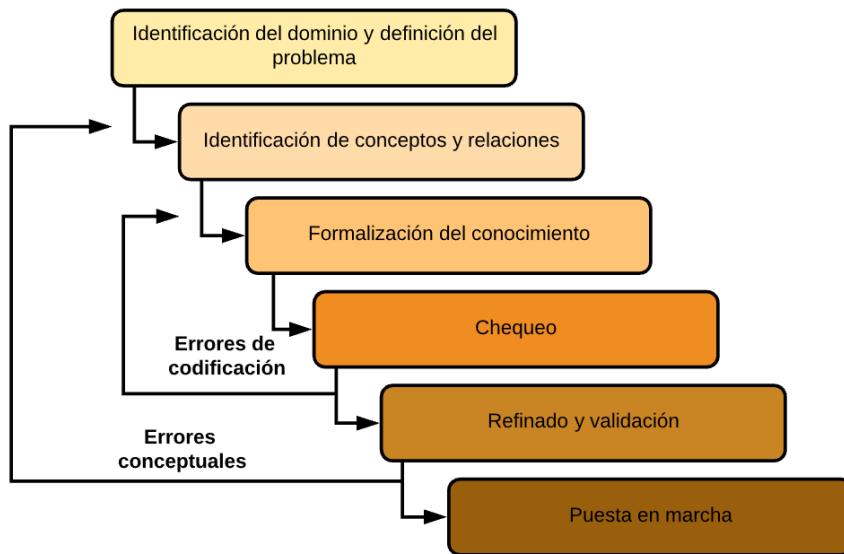


Figura 3.2: Fases de la adquisición del conocimiento.

Por otro lado, una entrevista estructurada es aquella donde el contenido, cuestiones y demás aspectos si están predefinidos. De esta forma se asegura el conseguir una adquisición de conocimiento más sistemática, además de tener un mayor control sobre el alcance del dominio del sistema.

Habitualmente, las primeras entrevistas realizadas con el experto suele ser informales, con el objetivo de que el ingeniero del conocimiento establezca el marco del dominio del conocimiento, además de conocer la forma de proceder del experto. Una vez el ingeniero del conocimiento empiece a conocer diferentes nociones y conceptos del tema en cuestión, es recomendable comenzar a realizar entrevistas más estructuradas.

A la hora de redactar los resultados de una entrevista, es conveniente hacerlo de forma organizada, definiendo una estructura con diferentes apartados. Es conveniente que dicha estructura cuente con los siguientes apartados:

- Fecha, hora, lugar y asistentes en la entrevista.
- Fuentes de conocimiento.
- Soporte de la entrevista (presencial o plataforma usada).
- Objetivos a lograr en la entrevista.
- Modalidad de entrevista (estructurada o no estructurada).
- Planteamiento de la sesión.
- Resultados de la sesión.
- Plan de análisis y sus resultados.
- Tiempo y recursos empleados en la entrevista.

### 3.4 Sistema experto

Un sistema experto es un tipo de sistema basado en el conocimiento creado para solucionar un problema emulando la mente de una persona que sea experto en un determinado campo. Como se ha visto en la anterior sección, la ingeniería del conocimiento es la encargada de diseñar y desarrollar sistemas expertos.



Figura 3.3: Clasificación de los sistemas expertos.

Los sistemas expertos, como todos los SBC, cuentan con los siguientes componentes: [GT16]

- *Base de conocimientos*: contiene el conocimiento representado capaz de resolver problemas sobre el tema en cuestión.
- *Base de hechos*: contiene la definición del entorno sobre el que se van a resolver problemas.
- *Motor de inferencias*: se encarga de modelar el proceso de razonamiento humano, contrasta con la base de conocimiento los hechos dados por el usuario para deducir nuevos hechos.

Otro componente es la interfaz, la cual permite la interacción entre el sistema y el usuario. Su diseño es especialmente importante. Una interfaz interactiva y fácil de usar hará que el sistema sea más atractivo para el usuario.

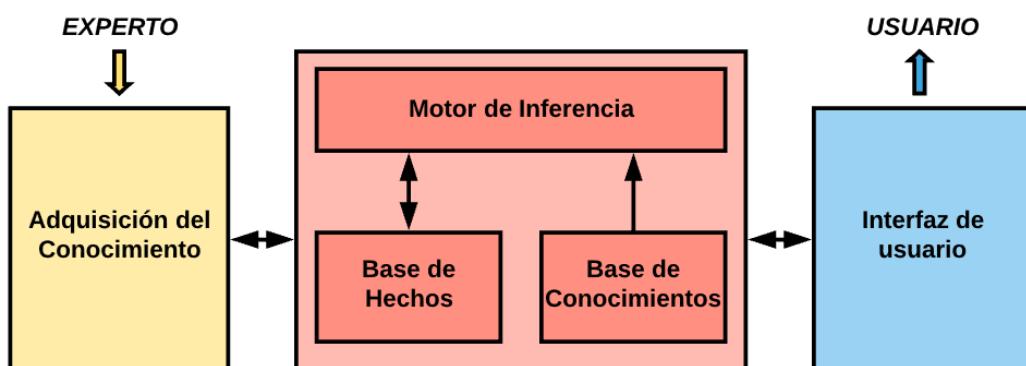


Figura 3.4: Estructura de un sistema experto.

### 3.5 Data lake

Un data lake es un repositorio centralizado que permite almacenar, compartir, gobernar y descubrir datos tanto estructurados como no estructurados a cualquier escala. Los data lakes no requieren de un esquema predefinido, ya que se pueden guardar y procesar datos sin tener un esquema y en cualquier formato, sin conocer como se van a explotar estos datos en el futuro.

Su uso principal es el almacenamiento de grandes cantidades de datos en su formato original, sin aplicar agregaciones ni transformaciones, estén estos datos estructurados o no estructurados. Además, aporta herramientas para su posterior procesamiento y administración. Esto se debe al auge de tecnologías de big data [MT16], que permiten analizar datos que antes eran imposibles de relacionar, de tal forma que en un data lake se pretende almacenar todos los datos de una organización o proyecto, aunque no tengan uso conocido.

El principal beneficio de la implementación de un data lake es soportar que los datos no vayan al proceso, si no que el proceso vaya a los datos. Además de tener mayor rendimiento para la optimización de procesos, al poseer información más reciente y precisa. Para crear un data lake no existe una metodología clara, pero es recomendable considerar los siguientes pasos:

- *Adquisición de datos*: consiste en la obtención de los datos del data lake. Se debe de identificar las fuentes y los conjuntos de datos que son de mayor valor para un determinado proceso.
- *Grooming Data*: es el conjunto de procesos por los que los datos son transformados en datos consumibles para posibles usos futuros. Suelen ser cosas sencillas, como transformar un fichero CSV en una matriz, normalizar los datos o generar datos derivados aplicando técnicas de inteligencia artificial.
- *Provisión de datos*: conjunto de procesos que permiten acceder a los datos del data lake de acuerdo con las políticas que este tenga establecidas.
- *Preservación de los datos*: políticas que determinan qué datos deben de conservarse y cuáles no.

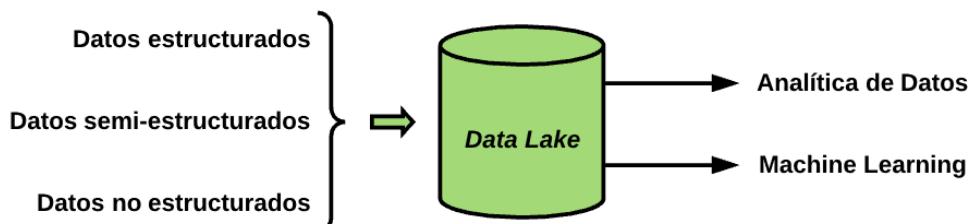


Figura 3.5: Esquema de un data lake.

### 3.6 Knowledge Discovery in Databases

En el año 1996, el científico de datos Usama Fayyad, [FPS96] definió el término Knowledge Discovery in Databases como el proceso no trivial de identificar patrones válidos y usables a partir de conjuntos de datos. Por tanto, el Proceso KDD tiene el objetivo de averiguar la naturaleza, cualidades o relaciones entre los elementos que forman un conjunto de datos.

Este hecho hace que el KDD sea un proceso habitualmente largo, dependiendo del dominio del problema y el propósito. Por ello, este proceso se divide en diferentes fases, cuyo seguimiento es crucial para obtener conocimiento de alta calidad. Este proceso es iterativo e interactivo. Por iterativo se entiende que la estructura temporal no tiene porque seguir una progresión lineal, por lo que terminar una fase puede tanto requerir avanzar a la siguiente fase o volver a alguna fase anterior para realizar alguna modificación. Por otro lado, por interactivo se ilustra la necesidad de la participación del usuario de forma activa a lo largo del proceso.

Las diferentes fases del Proceso KDD son las mostradas en la Figura 3.6, las cuales serán explicadas ampliamente en el capítulo referente a las metodologías usadas en el proyecto.

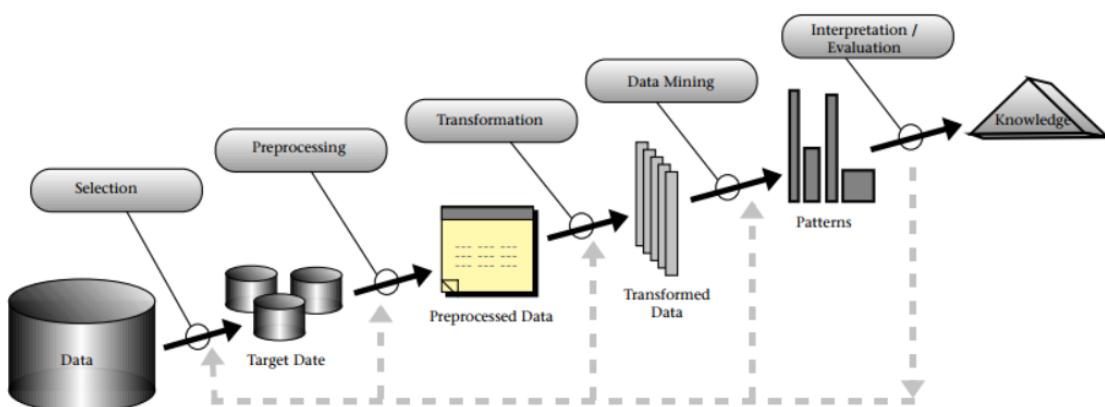


Figura 3.6: Fases del proceso KDD.

El KDD es un área que está tomando importancia en los últimos años debido al gran crecimiento actual de las bases de datos, y de la capacidad del hardware disponible para procesar los datos que contienen. Dentro del Proceso KDD, más específicamente en su etapa de minería de datos, son usados algoritmos de aprendizaje automático. En el caso de este proyecto, se hará uso tanto de algoritmos de aprendizaje supervisado como no supervisado.

#### Aprendizaje no supervisado

El aprendizaje no supervisado trata con datos o que no están etiquetados o cuya estructura sea desconocida. Por ende, los modelos de este tipo de aprendizaje, permiten explorar la estructura de los datos para extraer información de ellos sin contar con una variable como

guía. Existen técnicas de aprendizaje no supervisado no solo para descubrir estructuras en los datos no etiquetados, si no que también puede ser utilizado para comprimir datos.

Un ejemplo de ello es la reducción de dimensionalidad. Habitualmente se trabaja con datos con alta dimensionalidad, lo que puede presentar un reto en el caso de contar con un espacio de almacenamiento limitado, además de poner a prueba la capacidad computacional de los algoritmos. Esta reducción de dimensionalidad se utiliza a la hora de preprocesar las diferentes características de un conjunto de datos, con el objetivo de eliminar el máximo ruido posible, comprimiendo los datos en un espacio dimensional más pequeño que mantenga la máxima información posible. Este proceso hay que hacerlo con la máxima precaución, ya que también puede degradar el rendimiento de los algoritmos.

La reducción de dimensionalidad también es usada para facilitar la visualización de datos. De esta forma un conjunto de datos que cuente con gran dimensionalidad, podrá ser proyectado en espacios de una, dos o tres dimensiones a través de los gráficos o los histogramas que sean necesarios.

Los modelos de aprendizaje no supervisado también tienen la capacidad de dividir los datos en diferentes grupos, denominados clusters. El clustering es una técnica de análisis de datos que permite organizar un montón de información en diferentes grupos sin tener información previa de ellos. [Con12] Cada uno de estos grupos contendrá elementos semejantes entre ellos y que disten de los elementos de los demás grupos. Esta técnica es muy recomendable de llevar a cabo cuando el objetivo es estructurar la información y conocer relaciones existentes entre los datos. Un ejemplo de algoritmo de clustering es el algoritmo K - means.

Este algoritmo intenta dividir los datos en K clusters. Para ello, establecerá K centroides en el espacio de datos, para posteriormente ir elemento a elemento estableciendo a qué cluster pertenece cada uno. En concreto, cada elemento irá a parar al cluster cuyo centroide esté situado más cerca del propio elemento. En principio, no se conoce cual es el valor K que mejor divida los elementos, ya que dependerá precisamente de los propios datos con los que se este trabajando.

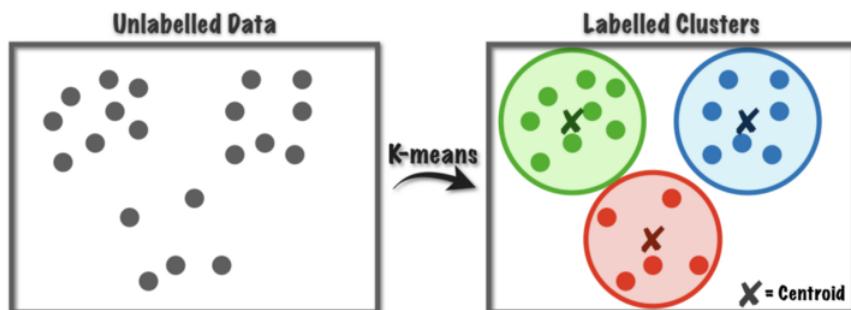


Figura 3.7: Ejemplo de uso del algoritmo K - means.

### 3. ANTECEDENTES: INTELIGENCIA ARTIFICIAL APLICADA A LA MEDICINA

A la hora de usar algoritmos de clustering, es recomendable normalizar los datos, para que los valores de cada atributo estén en escalas similares. Esta acción ayudará al clustering a crear grupos compensados y lógicos, ya que en estos algoritmos los grupos se forman a partir de distancias. Si existen atributos con escalas muy diferentes, los que tengan mayor escala dominarán las distancias.

#### **Aprendizaje supervisado**

El aprendizaje supervisado es un conjunto de técnicas que permiten la creación de modelos de predicción, gracias al uso de conjuntos de datos de entrenamiento etiquetados. Los algoritmos de aprendizaje supervisado intentan modelar relaciones y dependencias entre las predicciones obtenidas y las características de entrada para poder predecir los valores de salida de nuevos datos, gracias a las relaciones que aprendió de los conjuntos de datos anteriores. El aprendizaje supervisado proporciona un camino para convertir los datos en información.

Los principales tipos de tareas del aprendizaje supervisado son la clasificación y la regresión. Los algoritmos de clasificación intentan clasificar los diferentes elementos de datos entre las clases categóricas existentes, basándose en observaciones pasadas.

Por otro lado, los algoritmos de regresión predicen un valor continuo basado en entradas pasadas, con el objetivo de encontrar las relaciones existentes entre las diferentes variables que permitan obtener una predicción.

Un algoritmo tipo de aprendizaje supervisado de clasificación son los árboles de decisión, que son una técnica de aprendizaje automático inductivo que permite identificar conceptos de los datos a partir de las características que los representan. Los métodos basados en árboles potencian los modelos predictivos con alta precisión y fácil interpretación. [Con12]

Un árbol de decisión muestra una estructura similar a un diagrama de flujo donde un nodo interno representa una característica de los datos, las diferentes ramas representan las reglas que marcan el flujo, y cada nodo hoja representa un resultado. [Ger17] A la hora de construir un árbol, se debe de seguir la siguiente estrategia:

Primero, asignar todos los datos al nodo raíz, y seguir el siguiente proceso con cada nodo terminal:

- Verificar si la cantidad de datos que posee es muy pequeña o todas sus instancias pertenecen a la misma categoría, en ese caso, el nodo será terminal y se debe de tomar el siguiente nodo.
- En caso de no ser terminal, definir cuales son los atributos que permiten particionar el nodo.
- Seleccionar el atributo que permite hacer la partición más óptima y generarla.
- Tomar el siguiente nodo.

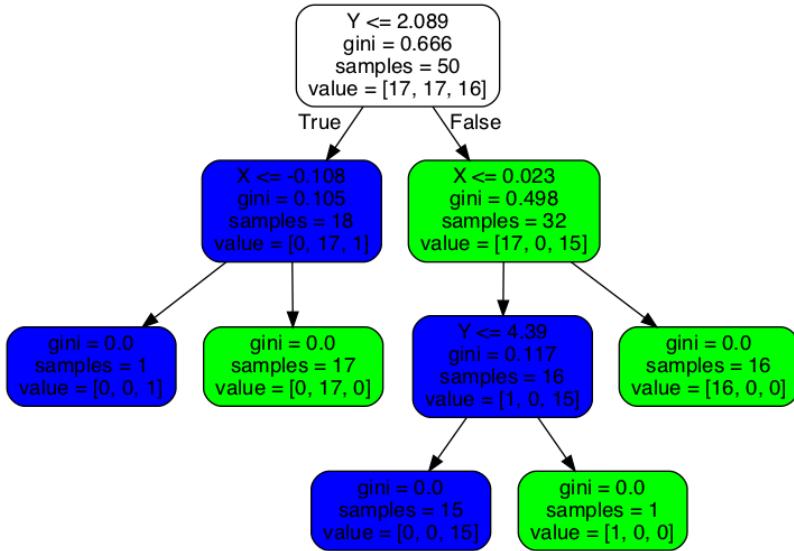


Figura 3.8: Árbol de decisión de la librería scikit-learn.

### 3.7 Web scraping

El web scraping es una técnica que sirve para extraer y almacenar datos de páginas web de forma automatizada, con el objetivo de analizarlos o utilizarlos para una determinada tarea. Es muy usado por empresas digitales las cuales se dedican a la recopilación de bases de datos. El proceso de web scraping se puede hacer de forma manual, el cual se basa en la idea del copiado y pegado manual de información de datos. Esta forma solo se pone en práctica cuando se desea encontrar y almacenar algún dato o información concreta, ya que es un proceso bastante laborioso. [Mit15]

Si se quiere encontrar y almacenar una cantidad de datos grande, será mejor optar por el web scraping automático, en el cual se hace uso de un software o un algoritmo que analiza diferentes páginas web para extraer información. Las dos maneras más habituales de realizar un web scraping automático son las siguientes:

- **Analizador sintáctico.** En términos de web scraping, los analizadores sintácticos se utilizan para convertir un texto en una nueva estructura. El analizador sintáctico, tras obtener el código HTML de la página web a analizar, podrá leer el contenido de dicha página para obtener la información y datos necesarios.
- **Bots.** Un bot es un software dedicado a realizar y automatizar determinadas tareas. En el caso de web scraping, son usados para recopilar datos tras la examinación automática de diferentes páginas web.

El uso del web scraping en el ámbito empresarial y financiero está muy generalizado, ya que permite obtener datos de contacto o información especial en un margen de tiempo muy pequeño. El objetivo de su utilización es obtener ventajas respecto a la competencia y conocer mejor las necesidades de sus clientes.

### 3.8 Estudios anteriores

Existen multitud de estudios y proyectos que hayan aplicado o estén aplicando diferentes técnicas de inteligencia artificial en diferentes ámbitos de la medicina. Un ejemplo actual es el proyecto desarrollado por el Instituto Tecnológico de Massachusetts en el año 2019, basado en un modelo de deep learning que puede detectar el riesgo de padecer cáncer de mama hasta cinco años antes de que aparezca, con solo examinar una mamografía.

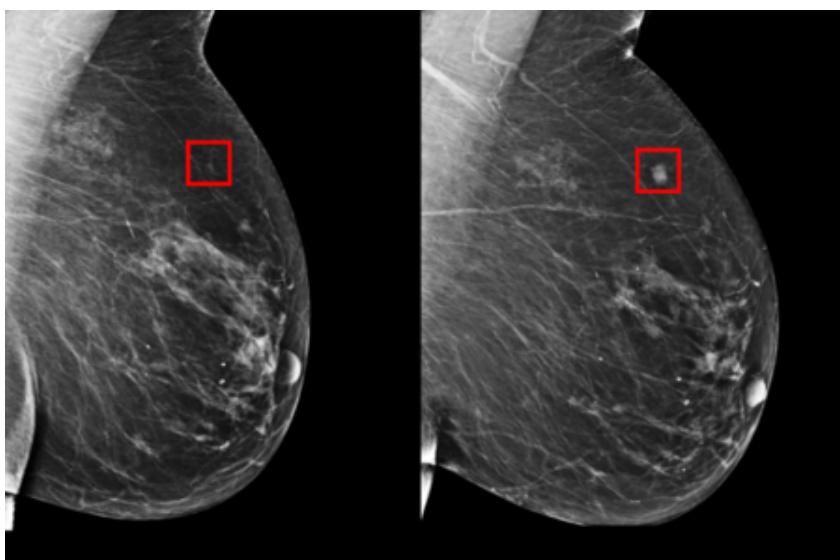


Figura 3.9: Ejemplos de cáncer de mama detectados en mamografías.

Otro ejemplo es el proyecto liderado por la Universidad de Boston centrado en la creación de un algoritmo que predice el riesgo de padecer Alzheimer con un acierto mayor que el de los neurólogos que participaron en el estudio, haciendo uso de la combinación de imágenes de la actividad cerebral, pruebas médicas y datos sobre edad y género de los pacientes del estudio.

Sin embargo, se retroceden unos cuantos años más atrás, hasta el 2000 concretamente, para poder hablar de un proyecto con relación directa con el desarrollado en este documento, INFODEC, un sistema basado en el conocimiento aplicado al diagnóstico y tratamiento extra-hospitalario en patologías infecciosas de vías respiratorias y otorrinolaringología. [Oli00]

El proyecto INFODEC contó con la financiación del Fondo de Investigación Sanitaria, dentro del plan nacional de I+D. Para comprobar su correcto funcionamiento, se puso a prueba en diferentes centros sanitarios de municipios de la comunidad autónoma de Asturias, tales como el Hospital Valle del Nalón, el Centro de Salud La Calzada y el Hospital de Caridad Jove.

El objetivo del proyecto fue la realización de un sistema computacional que ayudará a los médicos de atención primaria en los siguientes aspectos:

- Control de los hechos suministrados por el usuario.

- Apoyo a la decisión diagnóstica.
- Apoyo a la decisión terapéutica.
- Gestión de base de datos clínica.

El modelo lógico que sustenta el proceso de diagnóstico de enfermedades es el siguiente:

- Evaluar las hipótesis obteniendo una lista de las propuestas, ordenadas por su grado de credibilidad.
- Refinar el diagnóstico, seleccionando aquellas hipótesis que han superado cierto umbral de aceptabilidad.

INFEDEC presenta diferentes funcionalidades a sus usuarios, destacando la filiación del paciente, ya que permite introducir los datos del paciente para ser posteriormente almacenados en una base de datos. A partir de los hechos introducidos por el usuario, el sistema propone diferentes diagnósticos, ordenados por su grado de credibilidad. A partir de estos diagnósticos, INFEDEC presenta diferentes propuestas de tratamiento empírico, ordenadas también por un grado de adecuación.

Una vez se han realizado las tomas de decisiones que sean necesarias, INFEDEC permite almacenar la información en una base de datos habilitada para este fin.

Por último, los requisitos hardware para poder hacer uso de INFEDEC son contar con un ordenador que cuente con al menos 2 Mb de memoria RAM disponible y 5 Mb de disco duro libre, además de contar con un sistema operativo Windows 3.0 o superior. Actualmente, cualquier ordenador de los disponibles en el mercado podrá por tanto poner en marcha el sistema de ayuda a la decisión INFEDEC.



## Capítulo 4

# Metodologías y herramientas

DURANTE el desarrollo del proyecto, se hará uso de diferentes metodologías, con la finalidad de indicar cuales son los pasos a tomar para alcanzar los objetivos propuestos. Se presenta al lector en este capítulo unas nociones sobre estas metodologías a usar, para posteriormente ver su aplicación en el proyecto. Al contar este proyecto con tareas de diferentes ámbitos de la computación, es imposible a día de hoy encontrar una metodología que englobe a todas ellas, por lo que se hará uso de varías. Por último, se presentan las herramientas y medios más importantes empleados a lo largo del proyecto.

### 4.1 Metodologías ágiles

Las metodologías ágiles surgieron como alternativa a las metodologías tradicionales, las cuales destacan por su extrema rigidez, debido a diversos aspectos como la necesidad de tener los requisitos predefinidos antes de comenzar el proyecto, o la necesidad de seguir un conjunto de fases secuencialmente sin posibilidad de volver a atrás. Al ser estas metodologías secuenciales, la corrección de errores se vuelve más difícil, ya que estos errores no se descubren hasta una vez el proyecto ha finalizado. Hoy en día el entorno de desarrollo de software es demasiado inestable y cambiante por lo que estas metodologías no se adaptan.

En el año 2001, en una reunión convocada por Kent Beck, se originó el término Métodos Ágiles. Gracias a este término se formularon los siguientes postulados:

- Situar a los individuos y su iteración por encima de los procesos y las herramientas.
- Situar al software con correcto funcionamiento por encima de la documentación exhaustiva.
- Situar la colaboración con el cliente por encima de la negociación contractual.
- Situar las respuestas al cambio por encima del seguimiento de un plan.

Tras estos cuatro postulados, los presentes en la reunión redactaron 12 principios derivados de estos postulados, que reciben el nombre de los 12 principios del Manifiesto Ágil. [HV07]

- Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.

#### 4. METODOLOGÍAS Y HERRAMIENTAS

- Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.
- Entregamos software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto.
- Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.
- Los proyectos se desarrollan en torno a individuos motivados. Hay que darle el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.
- El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.
- El software funcionando es la medida principal de progreso.
- Los procesos ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.
- La atención continua a la excelencia técnica y al buen diseño mejora la agilidad.
- La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.
- Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.
- A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para a continuación ajustar y perfeccionar su comportamiento en consecuencia.

Algunas de las metodologías ágiles más utilizadas hoy en día son:

- *Extreme Programming*. Creada específicamente para promover la aplicación de técnicas de ingeniería para la creación de software. Su principal objetivo es que un equipo de desarrollo pueda producir software de mejor calidad de forma constante, y a su vez busca promover una buena calidad de vida para el equipo implicado. [Jos08]
- *Test Drive Development*. Es una técnica de ingeniería del software, cuya finalidad consiste en la realización de pruebas a lo largo del desarrollo software de un producto, generalmente unitarias. [Bec03]
- *Agile Project Management*. Hace referencia a un conjunto de metodologías para el desarrollo de proyectos que precisan de una especial rapidez y flexibilidad en su proceso. [Hig10]
- *Scrum*, que será usada en este proyecto y se profundizará en ella a continuación. Surgió como una metodología ágil de desarrollo software, pero en la actualidad se ha expandido a otras industrias.

## 4.2 Scrum

Scrum es una metodología ágil cuya base es la idea de creación de ciclos breves para el desarrollo de un proyecto, no necesariamente de desarrollo de software. Estos ciclos o iteraciones, en términos de esta metodología, reciben el nombre de sprint. Scrum define un ciclo de vida iterativo e incremental, mejorando la gestión de riesgos y aumentando la comunicación. [SB10]

En Scrum, la transparencia es clave, ya que todos los aspectos de los procesos que afecten al resultado deben de ser visibles para todos aquellos que administraron dicho resultado. También son dos pilares importantes la inspección y la revisión, ya que se debe de controlar con frecuencia los diversos aspectos del proceso para que puedan detectarse variaciones inaceptables en el mismo, además de comprobar que el producto está dentro de unos límites aceptables. Las principales características de esta metodología ágil son:

- Equipos de desarrollo organizados, de tamaño óptimo, y capaces de llevar a cabo diversas funcionalidades.
- El producto avanza en series de semanas o meses de longitud llamados sprints.
- Los requisitos son capturados como elementos en una lista denominada pila de producto, que será explicada más adelante.
- No se prescriben prácticas específicas de ingeniería.
- Usa reglas generativas para crear un entorno para la entrega de proyectos.

### Sprint

Cuando se hace referencia al término sprint, se está haciendo alusión a la parte más importante de la metodología Scrum. Sprint es el nombre que recibe cada uno de los ciclos o iteraciones que se deben de llevar a cabo dentro de un proyecto que use esta metodología. Un sprint va a permitir predefinir un tiempo de trabajo para diversas tareas, siendo la duración habitual de un sprint unas cuatro semanas o un mes. [Hon11]

En cada sprint, el resultado final son entregables o incrementos del producto, que aporten valor al cliente. La idea es que cuando se debe de afrontar un proyecto de larga duración, ese proyecto se pueda dividir en una serie de sprints que permita al equipo de trabajo dividir en el tiempo las tareas a llevar a cabo.

### Reuniones

A lo largo de un sprint, se llevan a cabo diferentes reuniones que ayudan al desarrollo del mismo.

- *Reunión de planificación de Sprint.* Reunión realizada al inicio del sprint. En ella se especifican los objetivos y las tareas a realizar durante el desarrollo del mismo.

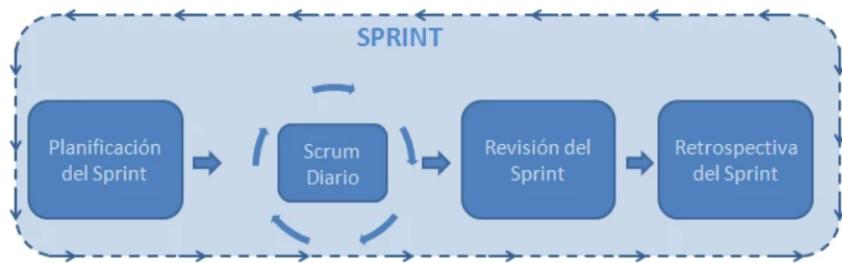


Figura 4.1: Reuniones de un sprint en Scrum.

- *Reuniones de Scrum diario.* Reunión de poca duración diaria con los miembros del equipo de proyecto. En ella cada miembro del equipo muestra los avances logrados respecto a la reunión anterior, especificando también sus objetivos para el próximo día, así como los problemas encontrados, en caso de que existan.
- *Reunión de revisión del sprint.* Realizada una vez ha finalizado el sprint. En ella se comprueba el progreso conseguido, comprobando cuales de los objetivos planteados al comienzo del sprint se han completado correctamente. Se debe de aceptar o denegar el sprint.
- *Reunión de retrospectiva.* Realizada después de la reunión de revisión del sprint y antes de la reunión de planificación del siguiente. Su objetivo es que el equipo de proyecto realice un análisis del desarrollo del proyecto hasta el momento, intentando poner en conocimiento del equipo posibles mejoras o fortalezas a afianzar.

## Roles en Scrum

Un equipo de desarrollo en Scrum suele contar con entre 3 y 9 miembros, y aparte, también existen los roles de Scrum Master y Product Owner. Lo ideal en un equipo de Scrum es que sus componentes sean independientes, con motivación propia, responsables y centrados en el usuario. Cada uno de estos roles tiene diferentes responsabilidades.

- *Product Owner.* Es el encargado de optimizar y maximizar el valor del producto. Debe de entender cual es la deriva que se desea para el producto durante el proyecto, para poder explicar a los stakeholders cual es el valor del producto en el que están invertiendo. Decide como será el resultado final, y el orden en el que se vayan realizando los diferentes sprints.
- *Scrum Master.* Tiene la responsabilidad de gestionar el proceso de Scrum, comprobando que se está llevando a cabo correctamente. Debe de revisar y validar la pila de producto e intermediar en las diferentes reuniones que se produzcan.
- *Equipo de desarrollo.* Formado por un grupo de profesionales que realizan las diferentes tareas asignadas a cada sprint. Los componentes del equipo deben de tener una buena capacidad de organización y gestión de tareas, para poder conseguir lograr los objetivos planteados en cada sprint de forma satisfactoria.

## Elementos de Scrum

Para la correcta realización de esta metodología ágil deben de existir diferentes artefactos de gestión que garanticen que el desarrollo de la metodología se está llevando a cabo correctamente. Estos artefactos son los siguientes:

- *Pila de producto*: es el inventario donde se almacenan todas las funcionalidades o requisitos en forma de lista priorizada. Dichos requisitos serán los que tendrá el producto o los que irá adquiriendo en sucesivas iteraciones. Esta lista debe de ser creada y gestionada por el cliente, con ayuda del Scrum Master. La pila de producto debe de ir evolucionando mientras el producto exista en el mercado.
- *Pila de sprint*: hace referencia a la lista de tareas que se planifican para realizar durante un sprint. En ella se asignan tareas a cada persona, además del tiempo necesario para finalizarlas. El objetivo de este componente es descomponer en unidades mas pequeñas el proyecto.
- *Incremento*: representa los requisitos que se han completado en un sprint.

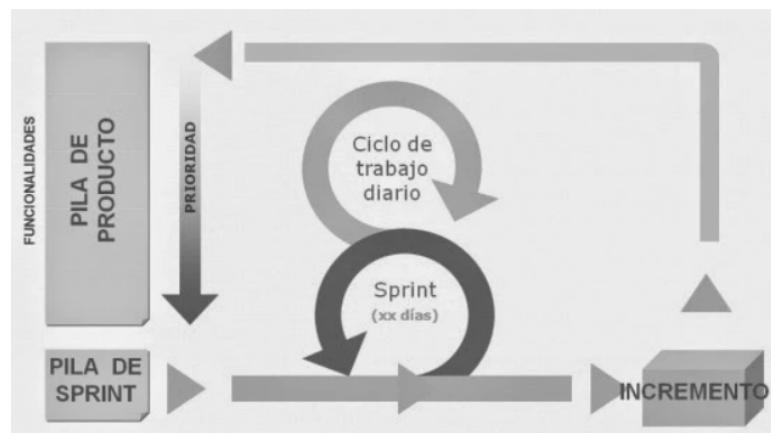


Figura 4.2: Ciclo de trabajo en Scrum.

## 4.3 Metodología IDEAL

La metodología IDEAL muestra la secuencia de pasos a llevar a cabo para construir un sistema experto haciendo uso de la ingeniería del conocimiento. Presentada por Alonso y Juristo, esta metodología cuenta con cuatro reglas o principios básicos: [Car87]

- *Regla de la Evidencia*. No se debe de aceptar nunca como verdadero lo que con toda evidencia, no se reconociese como tal. De esta forma, se evitará cuidadosamente la precipitación y los prejuicios.
- *Regla del Análisis*. Dividir cada una de las dificultades descubiertas en tantas partes como sea posible, para facilitar su solución, haciendo referencia al principio "divide y vencerás".

## 4. METODOLOGÍAS Y HERRAMIENTAS

- *Regla de la Síntesis.* Ordenar los conocimientos, empezando por los más sencillos, para poco a poco, elevarse hasta los más complejos. Esta regla expresa el principio de incrementabilidad en el diseño y construcción de los sistemas expertos.
- *Regla de la Prueba.* Hacer enumeraciones de tal forma que se pueda tener la seguridad de no haber omitido nada. La enumeración verifica el análisis, la revisión y la síntesis. De esta forma se evita dejar fuera cosas relevantes y el incorporar cosas no pertinentes.

En esta metodología se distinguen cinco grandes fases para llevar a cabo un sistema experto, que a su vez cuentan con una serie de etapas y pasos que se describirán a continuación. Sin embargo, cabe señalar que la metodología IDEAL es bastante flexible, de modo que siempre que el problema sea adecuado y esté perfectamente documentado, es posible saltar algún paso o etapa o incluso, alguna fase. En las siguientes subsecciones se exponen al lector las fases, con sus correspondientes etapas, que componen la metodología IDEAL. [GRB]

### Identificación de la tarea

Antes de iniciar el desarrollo de un sistema experto, se deben de definir los objetivos de la aplicación y determinar si la tarea puede ser tratada mediante la ingeniería del conocimiento. Las etapas a seguir durante esta fase son las siguientes:

- *Plan de requisitos y adquisición de conocimientos.* Para realizar el plan de requisitos es necesario haber empezado la adquisición del conocimiento, para que el experto aporte unas nociones básicas que sirvan para poder definir el funcionamiento y rendimiento deseado del sistema, sus limitaciones en tiempo y coste, requisitos de fabricación, etc.
- *Evaluación y selección de la tarea.* Se debe de realizar un estudio de viabilidad, haciendo uso del Test de Slagel, para poder evaluar la adecuación, plausibilidad, justificación y éxito del sistema experto a desarrollar.
- *Definición de las características de la tarea.* Se establecen y definen las características más relevantes del desarrollo de la aplicación, desde diferentes perspectivas, como pueden ser funcionales, operativas, de interfaz o de soporte. También se deben de establecer criterios de éxito, casos de prueba, etc.

### Desarrollo de prototipos

Una vez identificada la tarea, se desarrollan distintos prototipos que permiten definir y refinar las especificaciones del sistema, hasta conseguir especificaciones exactas de que se puede hacer y cómo. Las etapas para el desarrollo de un prototipo son las siguientes:

- *Concepción de la solución.* Se debe producir un diseño general del prototipo.
- *Adquisición del conocimiento y conceptualización.* Se debe de llevar a cabo la adquisición del conocimiento, modelando dicho conocimiento a través de la conceptualización.

- *Formalización del conocimiento.* Consiste en representar en el sistema todos los conceptos conseguidos en la anterior fase que se vayan a usar en el prototipo.
- *Implementación.* Se debe de implementar el conocimiento al sistema, esta etapa suele ser automática.
- *Validación y evaluación.* Haciendo uso de casos de prueba, se determina la fiabilidad del sistema experto a través de sus resultados.

### **Ejecución del sistema construido**

Tras añadir el conocimiento al sistema, se integra el sistema experto en sistemas generales con los que interactúen los usuarios. Para conseguirlo, se siguen los siguientes pasos:

- *Requisitos y diseño de la interacción.* Diseño de interfaces y puentes con los sistemas de hardware y software necesarios.
- *Implementación y evaluación de la integración.* Implementa la integración del sistema experto en el resto de sistemas.
- *Aceptación por el usuario del sistema final.* Prueba que pretende comprobar la satisfacción de los usuarios y los expertos con el prototipo.

### **Actuación para conseguir un mantenimiento perfectivo**

El prototipo debe de estar abierto a futuras mejoras y aumento de funcionalidades, permitiendo la incorporación de nuevos conocimientos. Las etapas a cumplir son las siguientes:

- *Definir el mantenimiento del sistema,* haciendo uso de técnicas de sistemas de información.
- *Definir el mantenimiento de la base de conocimientos,* dedicando una etapa especial al estudio de este mantenimiento.
- *Adquisición de nuevos conocimientos,* diseñando protocolos para que en caso de aparecer nuevos conocimientos, puedan captarse y registrarse.

### **Lograr una adecuada transferencia ecológica**

Se debe de lograr una adecuada implantación del sistema mediante una adecuada transferencia tecnológica que permita eliminar las diferencias de manejo entre los desarrolladores del sistema y los usuarios que hagan uso de ella.

- *Sesiones de entrenamiento,* entre los diseñadores y los usuarios, para explicar las funcionalidades el sistema.
- *Crear una documentación exhaustiva sobre el sistema.*

## 4.4 Proceso KDD

El proceso Knowledge Discovery in Databases, que a partir de ahora será denominado proceso KDD, hace referencia al proceso metodológico y secuencial seguido para encontrar un modelo válido, útil y entendible que describa patrones de acuerdo a la información de la que disponemos. El proceso KDD requiere la evaluación e interpretación de patrones y modelos para tomar decisiones con respecto a lo que constituye conocimiento y lo que no. Las fases de este proceso son las siguientes: [THC<sup>+</sup>16]

### Comprendión del dominio

Obtener el conocimiento sobre el dominio del estudio, sus propiedades y limitaciones, ya que así se podrán establecer los diferentes objetivos a lograr. En este paso es cuando se conocen las fuentes de información más importantes y quienes tienen control sobre ellas. Es vital establecer con claridad los objetivos y límites, ya que en el caso de no hacerlo, será muy fácil perder el rumbo del proceso debido a la gran marea de datos existentes a día de hoy.

También es conveniente evaluar la situación actual del problema, evaluando los antecedentes y requisitos del mismo, tanto en términos de negocio como de la minería de datos. Se debe de evaluar el conocimiento previo acerca del problema, cual sería la cantidad de datos apropiada para resolverlo, y las ventajas que supone la aplicación de la minería de datos al problema.

### Selección de datos

Una vez recolectados los datos y los objetivos estén marcados en la fase anterior, se elige un subconjunto de estos datos, los que se consideren más validos y útiles dependiendo del objetivo, para llevar a cabo el estudio y después homogeneizarlos, ya que así serán fáciles de procesar y analizar.

Para llevar a cabo la elección del subconjunto de datos es importante explorar todos los datos encontrados, creando una estructura general en ella y verificando su consistencia, con el objetivo de asegurar su completitud y corrección.

### Limpieza y preprocesado de datos

Se llevan a cabo tareas que garanticen la utilidad de los datos y determinen la confiabilidad de estos. Para ello se llevan a cabo tareas de limpieza de datos, como eliminar el ruido, eliminar variables, etc. Otras operaciones típicas de esta fase son la eliminación de comas, tabuladores, caracteres especiales, espacios, etc.

Habitualmente los conjuntos de datos disponibles suelen estar incompletos por diversos motivos, como por falta de valores de atributos, la existencia de outliers o inconsistencias. Estos datos pueden entorpecer el proceso de minería de datos y conducir a resultados erróneos o poco fiables. Gracias al preprocesado se mejorará la calidad de los conjuntos de datos,

y a larga, los resultados del proceso KDD.

En esta fase también se puede llevar a cabo la integración de nuevos datos, debido a la creación de nuevos campos, nuevos registros o una fusión de diferentes conjuntos de datos.

### **Transformación de datos**

Consiste en mejorar la calidad de los datos usando métodos como la reducción o transformación de dimensionalidad para reducir el número de variables a tener en cuenta. También se puede llevar a cabo la eliminación de columnas que varían a la vez, como por ejemplo la fecha de nacimiento y la edad de una persona. El objetivo es contar con aquellas variables que aporten información, suprimiendo aquellas que no aporten nada, o que su información sea aportada por otra variable. Esta etapa también se conoce como reducción de datos.

### **Minería de datos**

Una vez los datos estén limpiados y preparados para ser usados, se lleva a cabo la minería de datos. La minería de datos implica el uso de técnicas de bases de datos tales como el aprendizaje automático o el análisis predictivo. Se puede desgranar la minería de datos en las siguientes etapas:

#### *Elegir la función de minería de datos*

Elección del paradigma apropiado de la minería de datos para conseguir los objetivos establecidos, como pueden ser clasificación, regresión o clustering.

- *Clasificación.* Se caracterizan por el valor cualitativo de la variable objetivo con la que se realizan las tareas propias de la minería de datos. Los métodos encargados de clasificar predicen la probabilidad de una observación de pertenecer a cada una de las categorías posibles.
- *Regresión.* El valor de la variable objetivo es cuantitativo, es decir, la solución del problema será representada por un valor continuo que será determinada por las diferentes entradas del modelo, en vez de estar registrada a un grupo posible de valores.
- *Clustering.* Los algoritmos de agrupamiento o clustering son un conjunto de técnicas para encontrar grupos en los datos, con el objetivo de encontrar grupos lo más distinto posible entre ellos. Es una técnica de aprendizaje no supervisado muy útil cuando se quiere obtener conocimiento de la estructura de los datos.

#### *Elección del algoritmo de minería de datos*

Se deben de seleccionar uno o más algoritmos para buscar patrones en los datos. Cada algoritmo es diferente a la hora de trabajar y obtener los resultados, por ello, se deben conocer las propiedades de los algoritmos, para escoger el más indicado. Para realizar de forma satisfactoria esta fase es útil plantearse las siguientes preguntas:

#### 4. METODOLOGÍAS Y HERRAMIENTAS

- ¿Cuál es el mejor algoritmo para buscar modelos y patrones con los datos que cuento?
- ¿Cuáles son mis parámetros y criterios de evaluación?
- ¿Coincide el algoritmo seleccionado con el objetivo general del proceso KDD?

##### *Aplicación de los algoritmos*

Se aplican los algoritmos a los datos seleccionados, limpiados y preprocesados en pasos anteriores. La aplicación de estos algoritmos tiene como objetivo la búsqueda de patrones de interés en una forma de representación particular o un conjunto de tales representaciones, como reglas o árboles de clasificación.

##### **Interpretación del conocimiento**

Se evalúan los patrones generados en el paso anterior y su rendimiento para verificar que se han cumplido los objetivos marcados. Los resultados deben de presentarse en un formato entendible. Por ello las técnicas de visualización toman un valor muy importante, dado que los modelos matemáticos, o sus descripciones en texto, pueden ser bastante complicados de entender por una persona que no tenga la formación necesaria para ello.

Llegados a este punto, es importante plantearse si el proceso ha evolucionado tal y como se esperaba. En caso negativo, se recomienda retroceder a las fases anteriores, y hacer los cambios que sean necesarios.

##### **Uso del conocimiento descubierto**

Tras comprender los resultados del proceso y sus implicaciones, se aplica el conocimiento encontrado para resolver problemas gracias a él. Si los resultados no son los esperados, es necesario volver a los pasos anteriores para realizar los cambios que sean pertinentes.

#### **4.5 Metodología de creación de interfaces centradas en el usuario**

La metodología para la creación de interfaces gráficas centradas en el usuario reúne los conceptos, principios y fundamentos del Diseño Centrado en el Usuario (UCD), incluyendo los conceptos de pensamiento de diseño y los principios de usabilidad y experiencia de usuario. [SNG16]

La metodología cuenta con los siguientes atributos y principios:

- El usuario interviene constantemente en el proceso asumiendo diferentes roles y diversos grados de incidencia.
- Las actividades que realiza el usuario, en su contexto, es elemento funcional esencial que debe soportar la interfaz gráfica.
- Medición constante de la experiencia del usuario y la usabilidad de la solución teniendo en cuenta las características de eficiencia, eficacia, efectividad y satisfacción.



Figura 4.3: Bases de la metodología propuesta.

- Todas las ideas de solución son válidas hasta que el usuario objetivo y el propósito de la actividad demuestren lo contrario.
- La solución asume restricciones por las limitaciones de la tecnología en el proyecto, pero debe de procurar impulsar innovaciones desde su diseño.
- Existe una serie de requerimientos mínimos que debe seguir el diseño desde el inicio del desarrollo.
- El desarrollo de las fases de la metodología es iterativo e incremental.

Las fases a seguir en esta metodología son las mostradas en la Figura 4.4, y que analizaremos a continuación.

### Fase de estructuración

Para lograr una buena ejecución de la metodología, es necesario que exista una planificación de las actividades a realizar y una definición de elementos de trabajo. Por ello, en esta fase se estructura la estrategia a seguir de tal forma que permita ejecutar y completar todas las fases de la metodología. La finalidad de esta fase es establecer los procesos de gestión y los elementos de trabajo para la ejecución de las diferentes fases de la metodología. Considerando lo anterior, es conveniente realizar los siguientes pasos:

- Analizar los antecedentes del proyecto de desarrollo de software.
- Definir el tiempo necesario para construir la interfaz.
- Establecer el equipo de trabajo y fijar recursos y materiales.

#### 4. METODOLOGÍAS Y HERRAMIENTAS

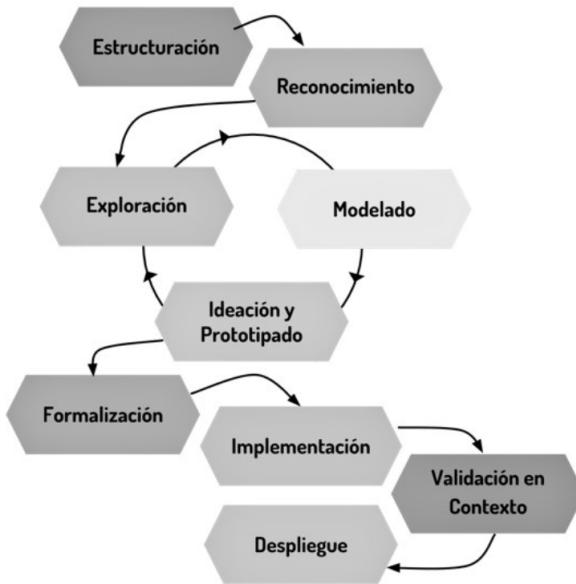


Figura 4.4: Fases de la metodología de interfaces gráficas centradas en el usuario.

- Identificar los espacios de trabajo y construir la estructura de trabajo.

El resultado de esta fase es un conjunto de tendencias o patrones que guíen todas las futuras fases de la metodología.

#### Fase de reconocimiento

Para llevar a cabo la construcción de una interfaz centrada en el usuario, es vital reconocer y caracterizar a los usuarios de la misma. Esta fase consiste en realizar un estudio del usuario para entender sus necesidades y su entorno. El objetivo es establecer un conjunto de características del usuario, tanto cognitivas, como físicas o emocionales, de las actividades y tareas que realiza en las que interviene la interfaz. A partir de esto, se proponen las siguientes etapas en esta fase:

- Conocer y caracterizar a los usuarios.
- Definir sus necesidades y problemas.
- Establecer el contexto de uso.
- Determinar el conjunto de actividades que realiza el usuario en función de su objetivo.
- Realizar la caracterización del usuario, contexto de uso y actividades.

La salida de esta fase es la caracterización del usuario objetivo, el contexto de uso y las actividades que necesita desarrollar por medio de la interfaz.

#### Fase de exploración

Además de conocer a los usuarios, también surge la necesidad de interactuar con ellos para conocer con profundidad cómo realizan las tareas y los retos a los que se enfrentan.

El objetivo en esta fase es analizar la ejecución de las actividades que realiza el usuario para definir elementos claves requeridos en la construcción de la interfaz. Se plantean los siguientes pasos:

- Observar un conjunto de usuarios objetivos realizando la actividad para identificar intuiciones sobre elementos relevantes a tener en cuenta en el desarrollo de la interfaz.
- Recibir las opiniones, sugerencias o problemas que el usuario afronta desde su experiencia para realizar la actividad.
- Identificar factores determinantes en el desarrollo de la actividad como patrones, elementos claves, conductas, comportamientos o protocolos, entre otros.
- Analizar y obtener conclusiones de las observaciones realizadas.

El resultado final de esta fase es un análisis de los factores determinantes en la actividad, junto con las características y elementos a considerar en el desarrollo de la interfaz.

### **Fase de modelado**

Tras comprender al usuario, es importante crear una coherencia sobre la información recolectada hasta el momento, es decir, se debe enmarcar el problema adecuadamente para crear una solución que sea correcta y válida. El objetivo de esta fase es procesar y sintetizar la información aportada por el usuario objetivo en la fase anterior, para poder crear relaciones y establecer patrones racionales del usuario. Se debe de construir un modelo de interfaz a partir de un concepto de diseño, que sirva para orientar las ideas y prototipos de la solución. El conjunto de etapas propuestas en esta fase son:

- Especificar los requerimientos del usuario y actividad para el desarrollo del modelo de interfaz.
- Identificar los componentes y elementos utilizados.
- Establecer la función y relación de los componentes identificados.
- Determinar acciones con los componentes y elementos para especificar posibles formas de interacción.
- Definir el modelo y alcance de la interfaz.

La salida de esta fase es un modelo de interfaz que incluya componentes, requerimientos, funciones y relaciones de la interfaz.

### **Fase de ideación y prototipado**

En esta fase se deben de llevar a cabo las máximas ideas posibles, con el fin de construir el prototipo ideal de la interfaz, a partir de la generación de múltiples ideas de solución. Se deben de llevar a cabo las siguientes etapas:

#### 4. METODOLOGÍAS Y HERRAMIENTAS

- Establecer un esquema que involucre los componentes del modelo para cada una de las ideas generadas.
- Definir la tipología del prototipo de cada idea planteada.
- Crear los prototipos y asegurar sus funcionalidades con respecto a la actividad realizada por el usuario.
- Evaluar el prototipo en contexto con usuarios objetivos.
- Concluir y refinar prototipos e ideas hasta obtener un conjunto reducido de prototipos ideales.
- Describir detalladamente los prototipos ideales y asignarles una prioridad a partir de la productividad del usuario para realizar la actividad.

El resultado de esta fase es un conjunto de prototipos de la interfaz a desarrollar.

#### Fase de formalización

Después de obtener un conjunto de posibles prototipos, se debe de realizar un proceso de formalización que considere la viabilidad técnica y tecnológica, utilizando indicadores que permitan evaluar y elegir la mejor solución posible. Se debe de evaluar la viabilidad de cada uno de los prototipos propuestos. El objetivo es formalizar el diseño final de interfaz que permita la interacción en propósito de la realización de las actividades por parte del usuario. De esta forma, las etapas a tener en cuenta en esta fase son las siguientes:

- Identificar cuales de las tecnologías actuales son más indicadas para la implementación de los prototipos.
- Realizar un mapeo entre las tecnologías y los prototipos, para definir que prototipos son viables y convenientes.
- Validar junto al usuario que prototipos son los más indicados, y posibles mejoras de los mismos.
- Describir formalmente la interfaz final.

El resultado final es la descripción formal de la interfaz a desarrollar, con todos los elementos de aspecto, estilo y lenguaje para su implementación.

#### Fase de implementación

A partir de la descripción formal conseguida en la anterior fase, se debe de realizar la implementación de la solución. El objetivo es implementar la solución final para realizar la validación en contexto. Las etapas propuestas para cumplir con este objetivo son:

- Implementar la interfaz siguiendo una metodología para la implementación de la misma.

- Realizar una validación funcional de la interfaz implementada.

El resultado es la implementación funcional de una interfaz que tenga en cuenta todas las tendencias de la descripción formal donde los usuarios objetivos especificaron las necesidades a cubrir por dicha interfaz.

### Fase de validación en contexto

Con la finalidad de integrar la interfaz a la solución de software, es necesario evaluar si la GUI usada cumple con todos los requerimientos y necesidades especificadas en el modelo de la interfaz. Esta fase consiste en una evaluación del usuario objetivo para realizar los ajustes finales y últimos cambios para integrar la interfaz en la solución del software. Se debe de validar la usabilidad y la experiencia de usuario con la solución de interfaz propuesta en la actividad en contexto para la que se desarrolló. Se deben seguir los siguientes pasos:

- Definir el conjunto de pruebas y protocolos en función de la verificación en contexto de la interfaz propuesta.
- Realizar pruebas con diferentes grupos de usuarios, entre los que se incluyen usuarios objetivos y otro tipo de usuarios, para identificar y realizar posibles mejoras.
- Evaluar por medio de las pruebas junto con los usuarios la solución a ser integrada.
- Realizar los cambios finales que sean necesarios.

El resultado de esta fase es la interfaz final, validada con los usuarios objetivo, que tendrán que certificar que se cumplen todos los requisitos que ellos mismos propusieron.

### Fase de despliegue

Esta fase consiste en la integración de la solución de interfaz propuesta al desarrollo de software. El objetivo de esta fase es desplegar e integrar la interfaz a la solución final. Las etapas son:

- Apropiar la metodología de implementación en el proyecto de desarrollo de software.
- Establecer comunicación con el equipo de desarrollo para exponer y explicar las funcionalidades de la interfaz.
- Implementar conexiones e integración con los demás componentes de software.
- Validar la funcionalidad de la interfaz, en relación a la actividad del usuario y los demás componentes y funciones del software.

El resultado de esta fase es la interfaz construida finalmente integrada en el desarrollo de software.

## 4.6 Aplicación de las metodologías en el proyecto

Una vez explicadas cada una de las metodologías a usar en el proyecto, se expone la forma en la se aplicarán a lo largo del mismo. El proyecto está encuadrado dentro de la metodología SCRUM, haciendo uso de 3 sprints, con un trabajo de 300 horas aproximadamente.

### 4.6.1 Sprint 1

En el primer sprint del proyecto se llevará a cabo el comienzo del desarrollo de un sistema experto, haciendo uso de la metodología IDEAL. Para ello se trabajará en las diferentes etapas de dicha metodología, incluyendo la adquisición del conocimiento, realizando una serie de entrevistas que se le plantearán al experto.

Por otro lado, en el mismo sprint, se iniciará el proceso de análisis de datos, definiendo el data lake del cual dicho proceso partirá. Para ello se consultará al experto para conocer su opinión sobre los distintos conjuntos de datos que serán añadidos a dicho data lake, estudiando sus posibles usos futuros, con el objetivo de extraer conocimiento de los mismos.

Tarea a realizar a lo largo del Sprint 1	Horas estimadas
Propuesta del sistema	10
Estudio de viabilidad	15
Adquisición del conocimiento	20
Conceptualización y representación del conocimiento	40
Definición del data lake	15

Cuadro 4.1: Pila de sprint: Sprint 1

### 4.6.2 Sprint 2

Una vez realizado el sistema experto y definido el data lake sobre el que trabajar, se llevará a cabo un proceso KDD, con el objetivo de extraer conocimiento de los datos. Se debe de comprobar que el conocimiento extraído coincide con el conocimiento del sistema experto. De esta forma, se podrá corroborar que el sistema experto ha sido desarrollado correctamente. También se trabajará por otro lado en expandir el conocimiento del sistema, gracias al conocimiento extraído de los datos, que será embebido en dicho sistema.

Tarea a realizar a lo largo del Sprint 2	Horas estimadas
Compresión del dominio y selección de datos	18
Preprocesado de datos	35
Transformación de datos	7
Tareas de minería de datos	35
Embeber conocimiento extraído en el SBC	35

Cuadro 4.2: Pila de sprint: Sprint 2

### 4.6.3 Sprint 3

En el último sprint del proyecto se realizará una interfaz para el sistema inteligente de ayuda a la decisión, haciendo uso de la metodología de creación de interfaces centradas en el usuario. Además, tras finalizar la interfaz, el experto propondrá diferentes casos de prueba para evaluar el sistema en su completitud.

Tarea a realizar a lo largo del Sprint 3	Horas estimadas
Estructuración y reconocimiento	10
Modelado, ideación y prototipado	15
Formalización e implementación	20
Validación y despliegue de la interfaz	10
Evaluación del sistema en su completitud	15

Cuadro 4.3: Pila de sprint: Sprint 3

## 4.7 Herramientas

Se procede a dar una breve descripción de las herramientas más relevantes usadas a lo largo del desarrollo del proyecto.

### 4.7.1 CLIPS

CLIPS es una herramienta que aporta un entorno de desarrollo para producir y ejecutar sistemas expertos. CLIPS estructura el conocimiento en hechos y reglas, representando los hechos información sobre el entorno en el cual se pretende llevar a cabo el razonamiento, y por otro lado, las reglas muestran aquellos elementos que permiten que el sistema evolucione. A día de hoy este entorno de desarrollo soporta diferentes paradigmas de programación como la programación lógica, la programación imperativa y la programación orientada a objetos.

Fue desarrollado a lo largo de los años 80 por la NASA y puede integrarse con lenguajes como C o C++. De hecho, su nombre es un acrónimo derivado de C Language Integrated Production System. Además, en el caso del lenguaje de programación Python, existe una librería que permite integrar CLIPS en dicho lenguaje, llamada ClipsPy. [Caf21] ClipsPy permitirá la definición tanto de hechos como de reglas, o la carga de las mismas desde un archivo externo, para poder crear un entorno de razonamiento idéntico al de CLIPS.

La principal ventaja que ha llevado a CLIPS a ser el entorno de desarrollo de sistemas expertos más conocido, es su simplicidad, ya que es muy fácil representar el conocimiento haciendo uso de reglas de producción. Existen múltiples manuales que documentan como usar el entorno de desarrollo de forma satisfactoria, entre los que destaca el manual de referencia, que se puede encontrar en su propia página web. [cli15]

## 4. METODOLOGÍAS Y HERRAMIENTAS

### Funcionamiento de CLIPS

Los hechos en CLIPS pueden ser representados por ordered facts o deftemplate facts. Los ordered facts no tienen una estructura predefinida y por tanto presentan un formato libre. Los deftemplate facts por su parte si presentan una estructura predefinida, en la cual se definen una serie de campos o slots. Por otro lado, las reglas están formadas por dos partes, en la parte izquierda de la regla se definen las condiciones a cumplir, y en la parte derecha las acciones a realizar en caso de cumplimiento.

```
(deftemplate usuario
    (slot nombre)
    (slot edad)
    (slot sexo))

(defrule rule_1
    (usuario(edad ?edad) (sexo hombre))
    (test(< ?edad 19))
=>
    (assert(hombre_menor))
    (printout t "El usuario es un hombre menor de edad" crlf)
)

(deffacts hechos
    (usuario(nombre "Juan Garcia") (edad 9) (sexo hombre))
)
```

Listado 4.1: Definición de un deftemplate, una regla y un hecho en CLIPS

Tal y como se vayan añadiendo hechos, si se cumplen las condiciones para que se lance una regla, esta misma se colocará dentro de la agenda. La agenda de CLIPS es la responsable de ordenar las diferentes reglas que se activan según su relevancia y la estrategia de resolución definida. Para lanzar las reglas almacenadas en la agenda, se hace uso del comando run.

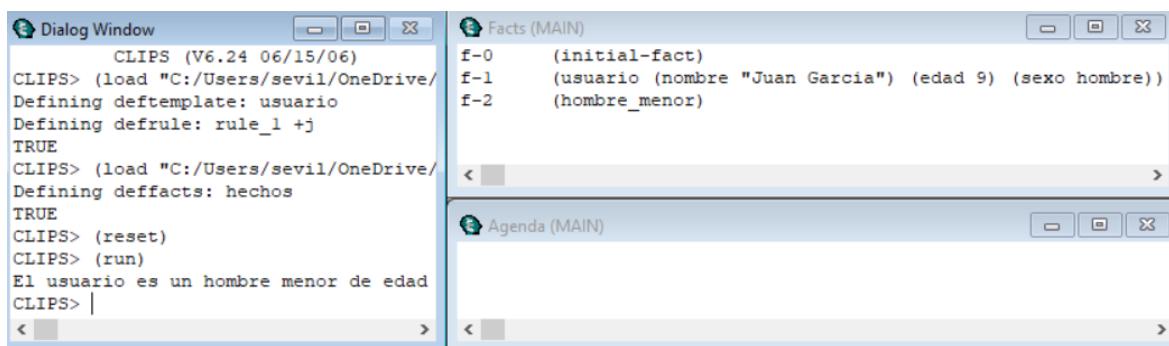


Figura 4.5: Ejemplo de ejecución de entorno de razonamiento en CLIPS.

En el ejemplo de la Figura 4.5 la agenda se encuentra vacía ya que la regla ya ha sido lanzada.

## Funcionamiento de Clipspy

Para ejecutar el mismo entorno de razonamiento mostrado en la Figura 4.5 haciendo uso de la librería Clipspy en Python, se debe de desarrollar el código del Listado 4.2. Como se puede comprobar, es prácticamente seguir los mismos pasos que en CLIPS pero plasmándolo en código de Python.

```
import clips

# Definimos el deftemplate usuario.
DEFTHEMEPLATE_STRING = """
(deftemplate usuario
  (slot nombre (type STRING))
  (slot edad (type INTEGER))
  (slot sexo (type STRING)))
"""

# Inicializamos el entorno de razonamiento.
environment = clips.Environment()

# Cargamos las reglas desde un archivo de CLIPS externo.
environment.load('constructs.clp')

# Cargamos la plantilla en el entorno.
environment.build(DEFTHEMEPLATE_STRING)

# Asignamos la plantilla a una variable.
template = environment.find_template('usuario')

# Creamos un hecho a partir de la variable de la plantilla.
fact = template.assert_fact(nombre = 'Juan Garcia', edad = 9, sexo = 'hombre')

# Hacemos correr la agenda.
environment.run()
```

Listado 4.2: Definición y ejecución del entorno de razonamiento con Clipspy

### 4.7.2 Google Colab

Google Colab es un servicio en la nube basado en los cuadernos Jupyter que permite el uso completamente gratis de GPU's ofrecidas por el propio Google. Es una herramienta ideal para practicar y mejorar los conocimientos en técnicas y algoritmos de ciencia de datos, con el objetivo de crear aplicaciones de aprendizaje automático.

Google Colab hace uso del lenguaje de programación Python en su versión 2.7 y 3.6, permitiendo así el uso de librerías esenciales para la ciencia de datos como Pandas, o Scikit-learn. Para hacer uso de esta herramienta, se deben de crear diferentes cuadernos en los que plasmar el código. Un cuaderno es un documento que contiene código ejecutable y también elementos de texto, lo que permite presentar trabajos de ciencia de datos de forma que el lector pueda interpretar el código con las respectivas explicaciones que el programador crea convenientes.

## 4. METODOLOGÍAS Y HERRAMIENTAS

Para ejecutar un cuaderno, es necesario conectarse a un entorno de ejecución, haciendo uso de una VM de Google Compute Engine. De inicio, el usuario cuenta con 12 GB de RAM y 50 GB de almacenamiento en disco disponibles para el uso en el entorno de ejecución, al cual se puede estar conectado un máximo de 12 horas.

### 4.7.3 Pandas y Scikit-learn

Pandas y Scikit-learn son dos librerías básicas de Python que permiten realizar estudios de ciencia de datos en dicho lenguaje de programación. Pandas es una extensión de la librería NumPy que permite el manejo de estructuras de datos. Permite leer y escribir ficheros en formato CSV de forma sencilla, además de ofrecer métodos para reordenar, dividir e indexar conjuntos de datos.

La librería Scikit-learn permite el uso del aprendizaje automático, incluyendo algoritmos de clasificación, regresión, clustering y reducción de dimensionalidad. La gran ventaja de usar esta librería es la gran variedad de módulos y algoritmos que facilitan el aprendizaje y trabajo del científico de datos en las primeras etapas de su desarrollo.

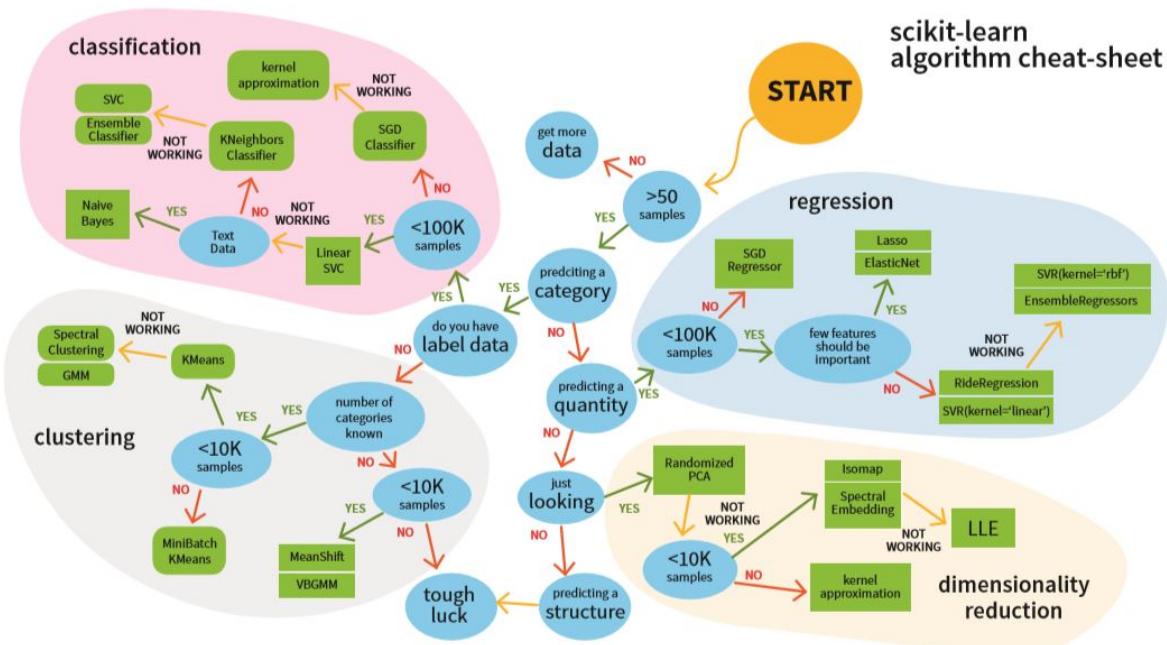


Figura 4.6: Diagrama de flujo para seleccionar el algoritmo a usar de Scikit-learn.

También se usará a lo largo del proceso de análisis de datos diferentes librerías que permitan crear gráficos, como Matplotlib o Seaborn, que permitirán la generación de gráficos a partir de datos contenidos en listas o arrays en el lenguaje de programación Python y su librería NumPy, por lo que podrán representar también datos contenidos en estructuras de Pandas, ya que es una extensión de dicha librería.

#### 4.7.4 PyQt

PyQt es la biblioteca gráfica de Qt para permitir el uso de sus herramientas en en lenguaje de programación Python. Qt utiliza originalmente el lenguaje de programación C++ de forma nativa, pero gracias a diferentes bindings, como PyQt, se puede hacer uso de Qt en otros lenguajes de programación. PyQt fue desarrollado por la compañía británica RiverBank Computing Ltd. [Qt21]

Qt es un framework orientado a objetos cuyo principal uso es el desarrollo de programas para crear interfaces gráficas de usuario. Es desarrollada como un software libre y de código abierto a través de Qt Project, donde contribuyen desde la propia comunidad hasta desarrolladores de empresas de prestigio como Nokia.

Para ayudar al usuario a crear interfaces de forma visual y por tanto, más cómoda para el mismo, Qt lanzó el programa Qt Designer. [Qtd21] El uso de este programa supone un gran ahorro de tiempo al usuario a la hora de desarrollar su interfaz, ya que no tendrá que desarrollarla de forma manual haciendo uso de algún lenguaje de etiquetas como XML, ya que Qt Designer crea el archivo XML correspondiente a la interfaz desarrollado visualmente por parte del usuario automáticamente.

#### 4.7.5 GitHub

GitHub es un servicio alojado en la nube mundialmente conocido elemental para desarrolladores de software, ya que permite gestionar el control de versiones del código alojado por los usuarios en diferentes repositorios.

En el caso de este proyecto, se ha creó un repositorio en esta plataforma (<https://github.com/sergiosb99/PedInf>) para almacenar todo el código desarrollado. Este repositorio ha permitido gestionar mejor las diferentes versiones del código, además de poder separar el trabajo en tres ramas diferentes, una por cada sprint.

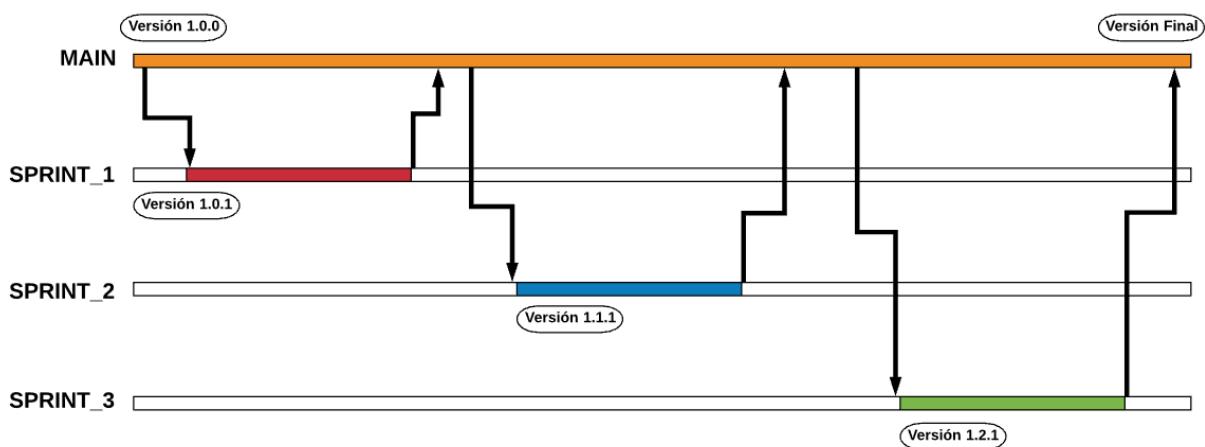


Figura 4.7: Control de versiones del proyecto.



## Capítulo 5

# Sistema Inteligente PedInf

**E**n este capítulo se exponen al lector los pasos llevados a cabo a lo largo del proyecto, así como sus resultados. Como se definió en la estrategia, en el proyecto se trabajará en el desarrollo de un SBC haciendo uso de la ingeniería del conocimiento, y por otro lado, se llevará a cabo un proceso de analítica de datos.

## 5.1 Sistema experto

La primera parte del proyecto consiste en el desarrollo de un sistema basado en el conocimiento, en concreto, un sistema experto que emule el razonamiento humano, actuando tal y como lo haría el experto en su situación.

### 5.1.1 Propuesta

Para empezar a desarrollar el sistema, es necesario describir su funcionalidad y finalidad, además de establecer su alcance y límites.

#### Descripción del sistema

El sistema experto que va a ser desarrollado en este proyecto, se basa en el diagnóstico de enfermedades infecciosas, y la posterior asignación de un tratamiento al paciente para combatir dicha enfermedad, siempre en niños de 0 a 13 años. Es decir, el sistema experto simulará la tarea de un pediatra, ya que le solicitará información al paciente para poder llegar a un diagnóstico y tratamiento certero.

El sistema no pretende sustituir a los pediatras, si no que el sistema quiere servir de complemento a los mismos, para facilitar y agilizar su tarea, además de aportar una visión diferente a la suya con la que poder contrastar su diagnóstico.

#### Alcance

El sistema abordará diferentes aspectos sobre las enfermedades infecciosas, como pueden ser:

- Síntomas generados por la enfermedad y gravedad que presentan.
- Rango de edad en el que suele manifestarse la enfermedad.

## 5. SISTEMA INTELIGENTE PEDINF

- Posibles tratamientos para paliar dicha enfermedad.

Además, el sistema necesitará la siguiente información del paciente:

- Nombre del paciente
- Edad del paciente
- Patologías previas

### Límites

El sistema no podrá detectar enfermedades infecciosas de personas con una edad superior a 13 años, ya que los diferentes aspectos que se van a tratar cambian drásticamente en la forma en la que se manifiestan en organismos superiores a los 13 años.

Además, el sistema se basará en la información correspondiente a los aspectos que se usarán para crear un diagnóstico, cualquier parámetro o información que se salga de esos parámetros, no podrá ser usada de una forma útil.

#### 5.1.2 Estudio de viabilidad

Para analizar como de viable es el proyecto, se realiza el Test de Slagel. En él, se califican diferentes aspectos para comprobar si el sistema a desarrollar es viable o por el contrario, no. El Test de Slagel considera cuatro dimensiones: plausibilidad, justificación, adecuación y éxito.

- *Plausibilidad*: permite definir si se cuenta con los medios necesarios para poder abordar el problema con la ingeniería del conocimiento, analizando las características del experto y la tarea que lleva a cabo.
- *Justificación*: se evalúa la necesidad de resolver el problema, así como la inversión a realizar para resolver el mismo.
- *Adecuación*: se analiza si el problema entra en los estándares para poder ser resuelto con técnicas de ingeniería del conocimiento, como pueden ser el uso de entrevistas, seguimiento del proceso que lleva a cabo un experto al resolver el problema en cuestión o técnicas conceptuales.
- *Éxito*: se determina las probabilidades de poder construir el sistema satisfactoriamente, estudiando si el sistema experto será aceptado por los usuarios como una herramienta útil.

Se muestran en los Cuadros 5.1, 5.2, 5.3 y 5.4 la valoración de cada uno de los aspectos de las cuatro dimensiones. Además, a continuación se muestra una justificación de porque el sistema a desarrollar cumple con las características esenciales de las cuatro dimensiones.

## Plausibilidad

Categoría	ID	Peso	Valor	Característica	Tipo
EX	P1	10	10	Existen expertos.	E
EX	P2	10	8	El experto asignado es genuino.	E
EX	P3	8	9	El experto es cooperativo.	D
EX	P4	7	8	El experto es capaz de articular sus métodos, pero no categoriza.	D
TA	P5	10	9	Existen suficientes casos de prueba; normales, ejemplares, correosos, etc.	E
TA	P6	10	10	La tarea está bien estructurada y se entiende.	D
TA	P7	10	7	Sólo requiere de habilidad cognoscitiva.	D
TA	P8	9	8	No se precisan resultados óptimos si no sólo satisfactorios, sin comprometer el proyecto.	D
TA	P9	9	8	La tarea no requiere sentido común.	D
DU	P10	7	7	Los directivos están verdaderamente comprometidos con el proyecto.	D

Cuadro 5.1: Test de Slagel: Plausibilidad

### Características esenciales:

- Existen expertos: si, el experto acaba de finalizar sus estudios en enfermería.
- El experto es genuino: el experto no llega a ser genuino, pero sí que tiene experiencia, ya que ha realizado prácticas y ha vivido situaciones a las que se enfrentará el sistema experto.
- Existen suficientes casos de prueba: se pueden implementar sin problema muchos casos de prueba para probar el correcto funcionamiento del futuro sistema.

## Justificación

Categoría	ID	Peso	Valor	Característica	Tipo
EX	J1	10	8	El experto no está disponible.	E
EX	J2	10	6	Hay escasez de experiencia humana.	D
TA	J3	8	8	Existe necesidad de experiencia simultánea en muchos lugares.	D
TA	J4	10	10	Necesidad de experiencia en entornos hostiles, penosos y/o poco gratificantes.	E
TA	J5	8	7	No existen soluciones alternativas admisibles.	E
DU	J6	7	9	Se espera una alta tasa de recuperación de la inversión.	D
DU	J7	8	10	Resuelve una tarea útil y necesaria.	E

Cuadro 5.2: Test de Slagel: Justificación

## 5. SISTEMA INTELIGENTE PEDINF

### *Características esenciales:*

- El experto está disponible: el experto estará disponible casi siempre debido a la cercanía de este con el creador del sistema.
- Necesidad de experiencia en entornos hostiles: el experto ha trabajado en situaciones de este tipo, aunque no son necesarias para el desarrollo del sistema.
- No existen soluciones alternativas admisibles: computacionales, a día de hoy, existen alternativas que hacen uso de la inteligencia artificial.
- Resuelve una tarea útil y necesaria: la salud de una persona en su infancia es un aspecto básico para el desarrollo de la sociedad en su conjunto.

### Adecuación

Categoría	ID	Peso	Valor	Característica	Tipo
EX	A1	5	7	La experiencia del experto está poco organizada.	D
TA	A2	6	7	Tiene valor práctico.	D
TA	A3	7	7	Es una tarea más táctica que estratégica.	D
TA	A4	7	10	La tarea da soluciones que sirvan a necesidades a largo plazo.	E
TA	A5	5	9	La tarea no es demasiado fácil, pero es de conocimiento intensivo, tanto propio del dominio, como de manipulación de la información.	D
TA	A6	6	7	Es de tamaño manejable, y/o es posible un enfoque gradual y/o, una descomposición en subtareas independientes.	D
EX	A7	7	8	La transferencia de experiencia entre humanos es factible (experto a aprendiz).	E
TA	A8	6	9	Estaba identificada como un problema en el área y los efectos de la introducción de un SE pueden planificarse.	D
TA	A9	9	7	No requiere respuestas en tiempo real.	E
TA	A10	9	8	La tarea no requiere investigación básica.	E
TA	A11	5	7	El experto usa básicamente razonamiento simbólico que implica factores subjetivos.	D
TA	A12	5	9	Es esencialmente de tipo heurístico.	D

Cuadro 5.3: Test de Slagel: Adecuación

### *Características esenciales:*

- La tarea da soluciones que sirvan a necesidades a largo plazo: el sistema a desarrollar en este proyecto es una buena iniciativa para trabajar en el futuro con ella.
- La transferencia de experiencia entre humanos es factible: la medicina y la enfermería son temas que son fáciles de explicar entre humanos siempre y cuando el experto tenga la capacidad de dar el enfoque correcto.
- No requiere respuestas en tiempo real: no requiere de ella.

- La tarea no requiere investigación básica: se necesita ir un paso más allá, adentrarse en la medicina, para poder diagnosticar enfermedades.

## Éxito

Categoría	ID	Peso	Valor	Característica	Tipo
EX	E1	8	9	No se sienten amenazados por el proyecto, son capaces de sentirse intelectualmente unidos al proyecto.	D
EX	E2	6	7	Tienen un brillante historial en la realización de esta tarea.	D
EX	E3	5	7	Hay acuerdos en lo que constituye una buena solución a la tarea.	D
EX	E4	5	8	La única justificación para dar un paso en la solución es la calidad de la solución final.	D
EX	E5	6	7	No hay un plazo de finalización estricto, ni ningún otro proyecto depende de esta tarea.	D
TA	E6	7	10	No está influenciada por vaivenes políticos.	E
TA	E7	8	8	Existen ya SE que resuelvan esa o parecidas tareas.	D
TA	E8	8	8	Hay cambios mínimos en los procedimientos habituales.	D
TA	E9	5	9	Las soluciones son explicables o interactivas.	D
TA	E10	7	9	La tarea es de I+D de carácter práctico, pero no ambas cosas simultáneamente.	E
DU	E11	6	6	Están mentalizados y tiene expectativas realistas tanto en el alcance como en las limitaciones.	D
DU	E12	7	8	No rechazan de plano esta tecnología.	E
DU	E13	6	8	El sistema interactúa inteligente y amistosamente con el usuario.	D
DU	E14	9	8	El sistema es capaz de explicar al usuario su razonamiento.	D
DU	E15	8	9	La inserción del sistema se efectúa sin traumas; es decir, apenas se interfiere en la rutina cotidiana de la empresa.	D
DU	E16	6	7	Están comprometidos durante toda la duración del proyecto, incluso después de su implantación.	D
DU	E17	8	8	Se efectúa una adecuada transferencia tecnológica.	E

Cuadro 5.4: Test de Slagel: Éxito

### Características esenciales:

- No está influenciada por vaivenes políticos: el asunto que se trata en el sistema está totalmente al margen del mundo de la política.
- La tarea es de I+D de carácter práctico: el sistema basará su funcionamiento en la investigación de la información necesaria, cuyo objetivo será encontrar una serie de experiencias que plasmar en una serie de reglas.
- No rechazan de plano esta tecnología: en la actualidad existen múltiples de sistemas expertos que son usados en el mundo de la medicina para distintos cometidos.
- Se efectúa una adecuada transferencia tecnológica: el sistema podrá efectuar una buena transferencia tecnológica con el usuario.

## 5. SISTEMA INTELIGENTE PEDINF

### Evaluación de la aplicación

#### PLAUSIBILIDAD

$$VC1 = \prod_{i=1,2,5} \frac{Vpi}{Vui} \left( \prod_{i=1}^{10} Ppi * Vpi \right)^{1/10} = 1 * (5,16 * 10^{18})^{1/10} = 74,35$$

#### JUSTIFICACIÓN

$$VC2 = \prod_{i=1,4,5,7} \frac{Vji}{Vui} \left( \prod_{i=1}^7 Pji * Vji \right)^{1/7} = 1 * (8,67 * 10^{12})^{1/7} = 70,52$$

#### ADECUACIÓN

$$VC3 = \prod_{i=4,7,9,10} \frac{Vai}{Vui} \left( \prod_{i=1}^{12} Pai * Vai \right)^{1/12} = 1 * (2,35 * 10^{20})^{1/12} = 49,84$$

#### ÉXITO

$$VC4 = \prod_{i=6,10,12,17} \frac{Vei}{Vui} \left( \prod_{i=1}^{17} Pei * Vei \right)^{1/17} = 1 * (1,94 * 10^{29})^{1/17} = 52,83$$

Al hacer la media de los cálculos anteriores, el resultado es 61'89, siendo el máximo 76'21, la viabilidad de la aplicación es del 81'2 %. Por lo tanto, el sistema es viable y se puede seguir trabajando en él.

### 5.1.3 Adquisición del conocimiento

Una vez determinado tanto el alcance como los límites del sistema, y comprobado que el proyecto es viable, se procede a realizar el proceso de adquisición del conocimiento. Para ello, se llevó a cabo un total de tres entrevistas con el experto. Dichas entrevistas se pueden consultar en el Anexo A.

#### Entrevistas no estructuradas

La primera entrevista que se llevó a cabo con el experto, fue no estructurada, es decir, fue una entrevista donde el ingeniero del conocimiento no predefinió los temas a tratar ni cuestiones sobre conceptos en concreto. El objetivo del ingeniero del conocimiento en esta entrevista no fue extraer un conocimiento superlativo, si no empezar a conocer diferentes conceptos relevantes sobre las enfermedades infecciosas, con el objetivo de profundizar en esos conceptos en futuras entrevistas.

#### Entrevistas estructuradas

Tras la realización de la primera entrevista y empezar a conocer el dominio del tema en cuestión, la segunda y tercera entrevista pasaron a ser estructuradas.

En la segunda entrevista, el experto, pregunta sobre los conceptos tratados en la primera entrevista, para profundizar en ellos.

En la tercera entrevista, el experto y el ingeniero de conocimiento concertaron una lista de enfermedades infecciosas sobre las que hablar. La enfermedades tratadas en dicha entrevista son las siguientes: varicela, mononucleosis, escarlatina, gripe, bronquiolitis, sarampión, enfermedad de Kawasaki, tos ferina, enterobiasis, gastroenteritis y faringoamigdalitis.

### 5.1.4 Conceptualización

Una vez se han llevado a cabo las entrevistas, se deben captar los principales conceptos extraídos de las mismas. Estos conceptos serán los aspectos principales en los que se trabajará de aquí en adelante para poder modelar el sistema experto.

#### Glosario de conceptos

Se debe de dar a conocer los términos más importantes extraídos de la etapa de adquisición del conocimiento en un glosario, dando a conocer la definición de los mismos.

- *Síntoma*: altera el organismo de una persona para dar a conocer la existencia de una enfermedad.
- *Enfermedad infecciosa*: enfermedad causada por microorganismos patógenos como pueden ser bacterias, virus o parásitos.
- *Tiempo de incubación*: periodo de tiempo desde que el microorganismo causante de la enfermedad entra en el sistema del paciente hasta la aparición de síntomas.
- *Tiempo de exclusión*: periodo de tiempo que el paciente debe de estar alejado de otros niños por riesgo de contagio.
- *Formas de transmisión*: métodos mediante los que la enfermedad se transmite.
- *Tratamiento*: es el conjunto de medios que se usarán para acabar con una enfermedad o síntomas, puede incluir fármacos, antibióticos y otro tipo de medidas, como, por ejemplo, higiénicas y de descanso.
- *Medicamento*: fármacos integrados que rebajan los efectos de una enfermedad o molestia sobre el organismo del ser humano.
- *Patología previa*: enfermedad que un paciente posee previa entrada a una consulta médica, y que puede afectar a la hora de recibir un tratamiento por parte de un médico.
- *Patología incompatible*: enfermedad que anula el posible uso del medicamento del tratamiento, por lo que el tratamiento solo constará con otras medidas de higiene y descanso.
- *Diagnosticar enfermedad infecciosa*: proceso en el cual se comparan los síntomas que presenta el paciente y su edad, para comprobar si alguna enfermedad infecciosa genera esos síntomas y su edad entra en el rango de edad.

#### Tablas objeto – atributo – valor

Una vez conocidos los conceptos más importantes, se hace uso de tablas objeto – atributo – valor, que permitirán relacionar los conceptos identificados entre ellos, y las características de los mismos.

## 5. SISTEMA INTELIGENTE PEDINF

Objeto	Atributo	Valor
Paciente	Nombre	{Juan, Ana, Pedro}
	Edad	[0 – 12] años
	Síntomas	{Tos, Fiebre, Dolor de cabeza, Mucosidad}
	Patologías previas	{Alergia, Problemas corazón, Problemas respiratorios}

Cuadro 5.5: Tabla Objeto – Atributo – Valor: Paciente

Objeto	Atributo	Valor
Enf. Infecciosa	Nombre	{Varicela, Sarampión, Bronquiolitis}
	Síntomas	{Tos, Fiebre, Dolor de cabeza, Mucosidad}
	Rango de edad	[0 – 12] años
	Formas de transmisión	{Vía respiratoria, Microgotas, Saliva}

Cuadro 5.6: Tabla Objeto – Atributo – Valor: Enf. Infecciosa

Objeto	Atributo	Valor
Tratamiento	Medicamento	{Paracetamol, Penicilina, Amoxicilina, Mebendazol}
	Otras medidas	{Hidratación, Reposo, Ambiente húmedo}
	Patologías incompatibles	{Alergia, Problemas corazón, Problemas respiratorios}

Cuadro 5.7: Tabla Objeto – Atributo – Valor: Tratamiento

### Mapa de conocimientos

En el mapa de conocimientos se debe de reflejar el proceso de razonamiento que lleva a cabo el experto a la hora de generar un diagnóstico. En él se deben de identificar las decisiones que el experto realiza, haciendo uso de un esquema para representar dicho mapa. El mapa de conocimiento se representa en la Figura 5.1.

#### 5.1.5 Representación del conocimiento

En esta fase, se busca crear una correspondencia entre el dominio de la aplicación a desarrollar y un sistema de símbolos que use el SBC. En este proyecto se hará uso de reglas de producción para este fin. Estas reglas de producción se representan gracias al entorno de desarrollo que ofrece la herramienta CLIPS. Se muestra un ejemplo de estas reglas y de dos deftemplates en el Listado 5.1.

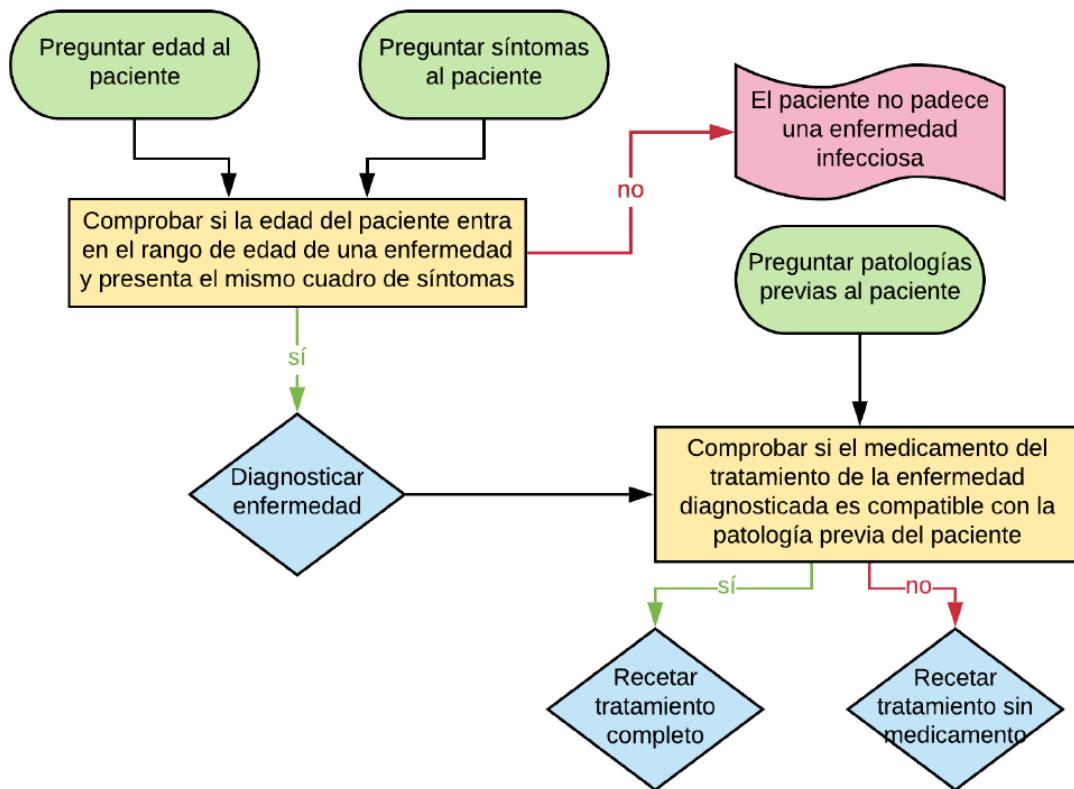


Figura 5.1: Mapa de conocimientos del sistema experto.

```

(deftemplate paciente
  (slot nombre)
  (slot edad)
  (slot patologia))

(deftemplate sintoma
  (slot nombre)
  (slot gravedad))

(defrule regla_1
  (paciente(edad ?edad))
  (sintoma(nombre picores)(gravedad fuerte))
  (sintoma(nombre piel_granosa)(gravedad leve))
  (sintoma(nombre piel_roja)(gravedad leve))
  (test(< ?edad 13))
=>
  (assert(varicela))
  (printout t "Enfermedad diagnosticada: Varicela." crlf)
)
  
```

Listado 5.1: Ejemplo de deftemplates y regla de producción

## 5.2 Análisis de datos

El segundo módulo del proyecto surge con la necesidad de corroborar el buen hacer del sistema experto desarrollado, además de poder conocer otros aspectos relevantes sobre las enfermedades infecciosas, estudiando cuales son los principales aspectos que favorecen la expansión de estas enfermedades. Para llevar a cabo este análisis de datos, se aplicará la metodología del proceso KDD sobre los conjuntos de datos que se consideren oportunos del data lake formado en este proyecto y que será presentado a continuación. Por lo tanto para empezar las fases de este proceso, se va a empezar presentando los diferentes conjuntos de datos con los que cuenta dicho data lake.

### 5.2.1 Presentación del data lake

Para llevar a cabo el proceso de búsqueda de datos públicos que permitieran llevar a cabo un proceso de análisis de datos en el proyecto, se tuvo en cuenta en todo momento el conocimiento del experto, recurriendo a él en varias ocasiones, debido a diversas dudas sobre la posible utilidad o importancia de algunos datos encontrados. Esta búsqueda también está influida directamente por los conceptos que el ingeniero de conocimiento aprendió y asumió como más relevantes durante la etapa de adquisición del conocimiento.

La cantidad de datos encontrada en este proceso fue muy numerosa. Por ello, se decidió formar un data lake, el cual permita almacenar todos los datos que sean necesarios y se estimen oportunos, independientemente del estado en el que se encuentren.

Por un lado se trajeron datos oficiales de la OMS, gracias a su repositorio de datos públicos. [GHO] Estos datos están organizados por países y tratan diferentes indicadores relacionados con la sanidad de los mismos.

Sin embargo, la OMS no ofrece datos sobre casos clínicos de enfermedades infecciosas, un aspecto vital para el desarrollo del proyecto, ya que de esta manera se podrá corroborar el buen hacer del sistema experto. Encontrar una base de datos de estas características no es tarea sencilla, ya que los datos de pacientes son privados y nunca deben de ser publicados por las autoridades sanitarias. Tras horas de búsqueda, se llega a una base de datos de casos clínicos de enfermedades anonimizados, es decir, sin datos personales del paciente, ni nombre, ni edad, ni sexo, etc.

Además también se han encontrado datos con intención de complementar a los ya nombrados. Estas bases de datos tratan cuestiones geográficas, económicas y climatológicas. También se han encontrado datos sanitarios relacionados con las enfermedades infecciosas en formato texto.

Con estos datos a priori se puede trabajar en las dos vertientes propuestas para el análisis de datos, ya que se han encontrado datos suficientes para llevar a cabo un estudio de cuales son los factores que favorecen la expansión de estas enfermedades en la sociedad, estudian-

do diferentes aspectos de la misma. Y por otro lado, se han encontrado casos clínicos que van a ayudar a diagnosticar enfermedades infecciosas a través de una serie de síntomas. A continuación, se profundiza en la descripción de cada uno de estos conjuntos de datos que posee el Data Lake.

## Datos Organización Mundial de la Salud

*Fuente de datos:* Repositorio de datos públicos de la OMS.

- *Fallecidos a causa de enfermedades durante la infancia:* se cuenta con siete base de datos, con datos desde el año 2000 hasta el 2017, que contienen el número de fallecidos desde los 0 hasta los 4 años como consecuencia de las siguientes enfermedades:
  - *Anomalías congénitas:* una anomalía congénita es aquella que se manifiesta desde antes del nacimiento, ya sea producida por un trastorno ocurrido durante el desarrollo embrionario, o como consecuencia de un defecto hereditario. Cada año 303.000 recién nacidos fallecen durante las primeras semanas de vida debido a anomalías congénitas.
  - *Asfixia:* se presenta cuando alguien no puede respirar debido a que un alimento, un juguete u otro objeto está obstruyendo las garganta o tráquea.
  - *Infecciones agudas en las vías respiratorias:* en esta base de datos se engloban todos los fallecidos por enfermedades infecciosas relacionadas con las vías respiratorias, lo que engloba un grupo de enfermedades infecciosas, entre la cuales podemos encontrar la neumonía, la otitis, o la sinusitis.
  - *Sepsis:* la sepsis es una respuesta contundente del sistema inmunitario cuando aparece una infección, comúnmente intestinal o del tracto urinario. Por tanto, en esta base de datos se muestran los fallecidos por una enfermedad infecciosa que afectó a los intestinos o al tracto urinario.
  - *Lesiones:* se presentan el número de fallecidos diversos a lesiones graves relacionadas con lesiones óseas o musculares.
  - *Meningitis y encefalitis:* la meningitis es una inflamación de las meninges del cerebro, causada por infecciones de oídos. Diversas cepas de bacterias pueden provocar meningitis bacteriana aguda. Por otro lado, la encefalitis es la inflamación del cerebro. Al igual que la meningitis puede ser causada por bacterias o virus.
  - *Malaria:* la malaria es una enfermedad infecciosa causada por parásitos, y transmitida por mosquitos al ser humano. Al año suelen morir 400000 personas por esta enfermedad aproximadamente, de las cuales, 78000 son niños.
  - *Sarampión:* el sarampión es una de las enfermedades infecciosas más comunes durante los primeros años de vida del ser humano, es causada por los virus de la familia paramyxoviridae.

## 5. SISTEMA INTELIGENTE PEDINF

- *VIH*: algunos niños suelen contraer el VIH al contraerlo de forma vertical, a través de sus madres VIH seropositivas. Puede ocurrir durante el embarazo, el parto o a través de la lactancia materna.
- *Diarrea*: las enfermedades diarreicas son la segunda mayor causa de muerte de niños menores de cinco años. Son enfermedades prevenibles y tratables.
- *Índices de vacunación*: estas ocho bases de datos muestran el índice de vacunación por país para la prevención de enfermedades como la poliomielitis, la hepatitis B o la difteria. Se poseen datos desde 1980 hasta 2019.
- *Casos diagnosticados de enfermedades infecciosas*: se cuenta con cinco bases de datos que indican el número de casos diagnosticados en cada país de las siguientes enfermedades infecciosas: cólera, malaria, meningitis, tuberculosis y VIH. Existen datos desde el año 1980 hasta 2019.
- *Delgadez extrema durante la infancia*: separado en dos bases de datos, ambas contabilizan la prevalencia de la delgadez extrema en niños. Una de estas dos bases de datos contabiliza esta prevalencia desde los 5 hasta los 9 años, mientras que la restante lo hace desde los 10 hasta los 19. Existen datos desde el año 1975 hasta 2016.
- *Anemia durante la infancia*: se posee una base de datos donde se aportan los datos de personas entre 0 y 5 años que sufren anemia. Datos desde el 1990 hasta el año 2016.
- *Ratio de mortalidad durante la infancia*: se cuenta con dos bases de datos que contabilizan la probabilidad de morir en edad pediátrica, calculando cuantos niños de cada 1000 fallecen. Una de las dos bases de datos contiene estos datos desde los 0 hasta los 5 años. Por su parte, la base de datos restante contiene los datos desde los 5 a los 14 años.
- *Lactancia materna exclusiva*: esta base de datos muestra el porcentaje de niños que han recibido lactancia materna durante los seis primeros meses de su vida.
- *Bajo peso al nacer*: se dispone de una base de datos donde se muestra la prevalencia de los niños con bajo peso al nacer. Datos desde el año 2000 al 2015.
- *Esperanza de vida*: se cuenta con esta base de datos, la cual muestra la esperanza de vida al nacer, y la esperanza de vida restante cuando una persona alcanza los 60 años de edad.
- *Muertes por la calidad del agua e higiene*: esta base de datos almacena los datos de las muertes atribuibles a la mala calidad del agua e higiene en personas de 0 a 5 años.
- *Contaminación del aire*: en esta base de datos se encuentra la concentración de partículas finas acumuladas en el aire, tanto en el ámbito rural, como en el urbano.
- *Infraestructuras sanitarias*: en esta base de datos se muestra el número de infraestructuras sanitarias por cada 100000 habitantes de hospitales y centros de salud. Datos

desde el año 2010 a 2013.

- *Personal sanitario*: en esta base de datos se exponen los datos del número de médicos, enfermeros y farmacéuticos por cada 10000 habitantes. Datos desde el año 2000 a 2019.
- *Camas de hospital por 10000 habitantes*: se dispone de una base de datos donde se muestra el número de camas de hospital por cada 10000 habitantes. Datos desde el año 2000 hasta 2017.
- *Gasto corriente en salud per cápita*: en esta base de datos se muestran los datos desde el año 2000 a 2018 del gasto medio en salud por persona llevado a cabo por cada país.

## Datos sobre cuestiones geográficas, económicas y climatológicas

*Fuente de datos*: Banco Mundial de Datos.

- *Extensión*: en esta base de datos se trata un aspecto geográfico, mostrando la extensión, medida en kilómetros al cuadrado, de cada país.
- *Población*: en esta base de datos se sigue trabajando con aspectos geográficos, en este caso, la población activa de cada país.
- *Producto interior bruto*: En esta base de datos se muestra el valor monetario de la producción de bienes y servicios de cada país. También denominado PIB, es una de las magnitudes macroeconómicas más utilizadas en la actualidad.
- *Temperatura*: se cuenta con un conjunto de datos donde se muestra la temperatura en grados centígrados diaria desde el año 1743 por cada país, ciudades y estados (si corresponde).
- *Precipitaciones*: esta base de datos muestra las precipitaciones en milímetros cuadrados por país entre los años 2012 y 2017.

## Datos en formato texto

*Fuente de datos*: Libros de casos clínicos

Se cuenta con archivos PDF que contienen datos sobre casos clínicos de diferentes enfermedades. En ellos se muestran diferentes aspectos como síntomas, tratamientos y factores a tener en cuenta de diferentes enfermedades.

## Datos sobre casos clínicos

*Fuente de datos*: Kaggle

Se ha hallado una base de datos donde se muestran casos clínicos de diferentes enfermedades, mostrando los síntomas que presentan los pacientes a lo largo de la enfermedad. Estos datos han sido sometidos a un proceso de anonimización, es decir, no se pueden relacionar

## 5. SISTEMA INTELIGENTE PEDINF

los casos clínicos a ningún paciente, al no contar la base de datos con ningún dato personal ni de contacto de los pacientes. Pose 41 enfermedades con un total de 4921 registros.

### Proceso de web scraping

El punto de unión entre la base de datos clínica y el conocimiento experto llegados a este punto se considera insuficiente, ya que solo dos enfermedades de las representadas en el sistema experto, existen también en la base de datos clínica, la varicela y la gastroenteritis. Se decide por tanto ampliar el punto de unión haciendo uso de web scraping. Gracias a esta técnica se podrá extraer datos de internet para crear bases de datos estructuradas. De esta forma, se podrá recopilar datos sobre las 9 enfermedades restantes recomendadas por el experto.

Inicialmente, para realizar el proceso se hará uso de web scraping automático, desarrollando un algoritmo para este fin. Sin embargo, dado el caso de falta de datos, o que los datos restantes por obtener sean escasos, se hará de forma manual, ya que al tratarse de una cantidad pequeña de datos, no supondrá ninguna pérdida de tiempo.

Para realizar el web scraping automático, se ha hallado una página web que aloja una base de datos sobre diferentes enfermedades infecciosas pediátricas, cuyo nombre es Pediatric Disease Annotations & Medicines [Ped17], abreviado, PedAM. Para poder visualizar datos de esta base de datos hay que realizar peticiones indicando el nombre de la enfermedad infecciosa deseada. No se permite la descarga de la base de datos en su totalidad.

De esta forma, consiguiendo el código fuente de las páginas web resultantes de las peticiones, se pueden conseguir datos referentes a los diferentes síntomas de una enfermedad. Para ello, se hace uso del algoritmo del Listado 5.2.

De las 9 enfermedades recomendadas por el experto (obviando la varicela y la gastroenteritis, que aparecen en la base de datos original), gracias a PedAM podemos extraer información de 7 ellas. Las únicas enfermedades no encontradas son la mononucleosis y la faringoamigdalitis. Los síntomas de estas dos enfermedades serán extraídas mediante web scraping manual. También se hará uso de esta técnica para la escarlatina y la enterobiasis, con el fin de ampliar los datos, ya que no se consideran suficientes. Al representar los datos buscados una cantidad pequeña, supondrá una ventaja realizar web scraping manual, ya que el tiempo que conlleva la confección de diferentes algoritmos para extraer dichos datos, será mucho mayor que si se realiza manualmente.

#### 5.2.2 Selección de datos

Tras describir todos los conjuntos de datos que contiene el data lake, llega el momento de seleccionar aquellos datos que vayan a ser útiles para el desarrollo del proyecto. Para ello, se especifica cuáles han sido los criterios para la selección de los mismos.

```

from bs4 import BeautifulSoup
import csv

def funcion (codigo_html):
    page = open(codigo_html,"r")
    soup = BeautifulSoup(page, 'html.parser')

    tabla_general = soup.find(id = 'D3')
    tablas = tabla_general.find_all('tr')

    nombre_enfermedad = tablas[1].find_all('td')
    tabla_sintomas = tablas[3].find(id = 'tdheight')
    sintomas = tabla_sintomas.find_all('a')

    for x in range(0,len(sintomas)):
        if x % 2 != 0:
            print(sintomas[x].text)
            registro = [[nombre_enfermedad[0].text,sintomas[x].text]]

            bbdd = open(r".\dataset.csv", 'a')
            with bbdd:
                writer = csv.writer(bbdd)
                writer.writerow(registro)

```

Listado 5.2: Algoritmo de web scraping usado

## Datos Organización Mundial de la Salud

- *Fallecidos a causa de enfermedades durante la infancia:* se han seleccionado los conjuntos de datos referentes a enfermedades infecciosas, que son los siguientes:

- *Infecciones agudas en las vías respiratorias*
- *Sepsis*
- *Meningitis y encefalitis*
- *Malaria*
- *Sarampión*
- *VIH*
- *Diarrea*

En el caso de la diarrea, esta puede ser provocada tanto por agentes infecciosos como por fallos en el colon. Sin embargo, tras poner en conocimiento del experto esta base de datos, él informa que al estar tratando con datos de humanos desde 0 a 4 años, es muy poco probable que un ser humano presente problemas de colon durante esa etapa, por lo que la mayoría de las defunciones por esta enfermedad en este intervalo de años, será provocado por infecciones.

- *Índices de vacunación:* la vacunación es un aspecto vital a la hora de frenar la expansión de enfermedades infecciosas.
- *Delgadez extrema durante la infancia:* la delgadez puede ser provocada debido a la

## 5. SISTEMA INTELIGENTE PEDINF

mala nutrición, mala higiene, etc. Estos factores pueden favorecer la expansión de bacterias o microbios.

- *Anemia durante la infancia:* la anemia puede aparecer por la falta de hierro y consecuentemente por una nutrición poco equilibrada, lo que puede disminuir las defensas del humano, siendo más duramente atacado por las enfermedades infecciosas.
- *Ratio de mortalidad durante la infancia:* conocer cual es el ratio de mortalidad habitual es un factor esencial para llevar a cabo el estudio.
- *Lactancia materna exclusiva:* la OMS afirma que el mejor alimento que puede recibir un recién nacido es la leche materna, al ser rica en hierro y proteger al niño de posibles infecciones.
- *Bajo peso al nacer:* contar con un bajo peso al nacer puede conllevar que el recién nacido aumente sus dificultades a la hora de enfrentarse a diferentes infecciones.
- *Muertes por la calidad del agua e higiene:* es vital para evitar la expansión de las enfermedades infecciosas la calidad del agua y su limpieza, ya que pueden ser un gran transportador de bacterias y hongos.
- *Contaminación del aire:* al igual que el agua, la contaminación del aire puede ser un vehículo para diferentes bacterias y hongos.
- *Infraestructuras sanitarias:* es de vital importancia contar con las infraestructuras sanitarias necesarias para poder tratar a cada paciente de forma correcta.
- *Personal sanitario:* al igual que las infraestructuras, es vital contar con un personal sanitario holgado para poder garantizar un trato adecuado a los pacientes.
- *Camas de hospital por 10000 habitantes:* contra más camas estén disponibles por 10000 habitantes, mejor atención se les podrá dar a los mismos.
- *Gasto corriente en salud per cápita:* es vital que un gobierno nacional invierta dinero en mejorar las prestaciones y servicios sanitarios.

### Datos sobre cuestiones geográficas, económicas y climatológicas

Finalmente tras valorar la posibilidad de seleccionar estos datos, no se cuenta con ellos, ya que se consideran que son datos con muy poca relación directa con el contagio de enfermedades infecciosas.

### Datos en formato texto

Han sido descartados todos ya que para los objetivos en los que se está trabajando, no se consideran útiles.

## Datos sobre casos clínicos

Tras realizar el proceso de web scraping para obtener datos sobre las enfermedades que se tratan en el sistema experto, el principal inconveniente es que la base de datos de casos clínicos, de las 41 enfermedades que posee originalmente, solo 18 de ellas son infecciosas. Sin embargo, representan un total de 2400 registros, por lo que se podrá trabajar en un futuro con esta base de datos para extraer conocimiento. Las 18 enfermedades son las siguientes: micosis, úlcera péptica, gastroenteritis, malaria, varicela, dengue, tifoidea, hepatitis (A, B, C, D y E), tuberculosis, resfriado común, neumonía, hipotiroidismo, hipertiroidismo e impétigo.

Las registros de las enfermedades restantes, son eliminadas, ya que no aportarán conocimiento útil. Por tanto se seleccionan los casos clínicos de enfermedades infecciosas y la base de datos construida a través del proceso de web scraping. El siguiente y último paso es adaptar los datos extraídos mediante web scraping a la base de datos clínica, para ello, se asociarán los síntomas hallados gracias a esta técnica con los de la base de datos clínica. En caso de no encontrar un síntoma semejante, se añadirá como síntoma nuevo.

Por tanto, tras terminar la selección de datos, el resultado es que se cuenta con una base de datos con la que trabajar para cada una de las dos vertientes que se llevaran a cabo. Por un lado, se posee una base de datos que nos permitirá investigar los factores de la expansión de estas enfermedades con 194 registros y 35 columnas, y por el otro, se cuenta con una base de datos con 3240 casos clínicos. Ambas están representadas en los Cuadros 5.8 y 5.9.

Disease	Symptom_1	Symptom_2	Symptom_3	...
Fungal infection	itching	skin_rash	nodal_skin_eruptions	...
Fungal infection	skin_rash	nodal_skin_eruptions	dischromic_patches	...
Fungal infection	itching	nodal_skin_eruptions	dischromic_patches	...
...	...	...	...	...
Pharyngotonsillitis	high_fever	cough	swollen_tonsils	...
Pharyngotonsillitis	cough	swollen_tonsils	mild_fever	...

Cuadro 5.8: Selección de datos: Casos clínicos

País	Total	Fallecidos	Media	Fallecidos	Total	Fallecidos	Media	Fallecidos	...
	Vías Respiratorias	Vías Respiratorias		Sepsis		Sepsis			...
Afghanistan	342325		19018.055556		132422		7356.777778		...
Albania	1709		94.944444		232		12.888889		...
Algeria	57438		3191.000000		34367		1909.277778		...
...	...		...		...		...		...
Zambia	150437		8357.611111		45546		2530.333333		...
Zimbabwe	101731		5651.722222		32402		1800.111111		...

Cuadro 5.9: Selección de datos: OMS

### 5.2.3 Preproceso y transformación de datos

Tras llevar a cabo la selección de datos, será necesario realizar un preprocesado para la limpieza de los mismos, con el objetivo de obtener tablas de datos preparadas para ser usadas en diferentes tareas de minerías de datos, con el objetivo de extraer conocimiento.

El objetivo del preprocesado de datos es mejorar la calidad de los datos, para posteriormente, haciendo uso de técnicas de extracción de conocimiento o minería de datos, poder obtener una mayor y mejor información.

Por otro lado, la transformación de datos busca mejorar la calidad de los datos haciendo uso de la reducción de dimensionalidad o realizando transformaciones como discretizar valores numéricos a categóricos.

A continuación se describen los pasos llevados a cabo en las etapas de preproceso y transformación de datos para cada una de las dos tablas de datos de las que se disponen en este punto del proyecto.

#### Tabla de datos clínica

Para poder trabajar con algún algoritmo de minería de datos cuya entrada sea esta tabla de datos, se debe de preprocesar, ya que en su estado actual no será posible realizar algoritmo alguno. Para ello, se llevan a cabo los siguientes pasos de preproceso:

- Crear un dataframe nuevo, con tantas columnas como síntomas existan y tantas filas como casos clínicos posea la base de datos clínica. De esta forma se obtiene una base de datos donde cada columna hace referencia a un síntoma y cada registro a un caso clínico.
- Rellenar todos los valores del dataframe nuevo con el valor 0 momentáneamente.
- Recorrer cada registro del dataframe antiguo, y por cada síntoma existente en el caso clínico, asignar valor 1 a la columna correspondiente a dicho síntoma en el dataframe nuevo.
- Borrar aquellos síntomas que no tengan valor 1 en algún registro. Es decir, se desechan en el dataframe nuevo aquellas columnas que hagan referencia a síntomas que no aparecen en ningún caso clínico.

Antes de terminar el preproceso, el experto recomendó borrar todas las enfermedades de hepatitis menos la de tipo A, debido a que esta es la predominante en niños, siendo poco habitual que un niño presente infección por el resto de tipos. También recomendó borrar algunos síntomas que él concibe como demasiado elevados para el tipo de aplicación que estamos desarrollando, como por ejemplo, el padecer un coma o contar con transfusiones de sangre anteriores. El resultado es una base de datos con 2760 registros y 68 columnas.

Disease	itching	skin_rash	nodal_skin_eruptions	continuous_sneezing	shivering	...
Fungalinflection	1	1	1	0	0	...
Fungalinflection	0	1	1	0	0	...
Fungalinflection	1	0	1	0	0	...
...	...	...	...	...	...	...
Pharyngotonsillitis	0	0	0	0	0	...
Pharyngotonsillitis	0	0	0	0	0	...

Cuadro 5.10: Preproceso de datos: Casos clínicos

**Tabla de datos OMS**

En el caso de la tabla de datos de la OMS, se han llevado a cabo diferentes operaciones.

- Calcular una nueva variable que englobe todos los casos de enfermedades infecciosas. Dicha variable será la media de niños fallecidos a causa por aquellas enfermedades infecciosas sobre las que OMS ofrece datos.
- Eliminar aquellas columnas relacionadas tanto con la media de fallecimientos como el total de fallecimientos de niños causados por diversas enfermedades infecciosas, es decir, las usadas para calcular la variable nueva.
- Tratamiento de valores vacíos. Para ello, primero se consulta qué columnas contienen valores vacíos para completarlos con la media del resto de valores de la columna.
- Redondear los decimales de los diferentes valores de la tabla de datos, dejando únicamente dos decimales por cifra.
- Normalizar los valores de la base de datos para que todos los valores estén en la misma escala.

En este caso se descarta realizar una reducción de dimensiones en este punto, ya que las variables con la que cuenta el dataframe no son excesivas. El resultado de llevar a cabo estos pasos de preproceso es la obtención de una base de datos con 194 registros y 21 columnas, plasmada en el Cuadro 5.11.

País	Vac. poliomielitis	Vac. antineumocó- cicas	Vac. sarampión 1	Vac. sarampión 2	...
Afghanistan	37.98	28.08	37.80	21.05	...
Albania	93.98	68.15	91.41	87.29	...
Algeria	78.54	25.54	73.29	87.33	...
...	...	...	...	...	...
Zambia	73.29	40.08	75.46	15.86	...
Zimbabwe	75.59	50.15	76.05	14.00	...

Cuadro 5.11: Preproceso de datos: OMS

### 5.2.4 Minería de datos e interpretación del conocimiento

Llegados a este punto, llega el momento de aplicar los algoritmos correspondientes sobre los datos. Como se ha definido a lo largo del proyecto, el análisis de datos cuenta con dos vertientes. La primera de ellas hace referencia al estudio de cuales son los motivos que permiten que las enfermedades infecciosas se expandan en nuestro planeta, haciendo uso de datos de la OMS. La otra vertiente hace referencia al estudio de casos clínicos de enfermedades infecciosas, para crear modelos de predicción, que nos ayuden a corroborar que el comportamiento del sistema experto desarrollado es correcto.

Para el primer cometido se hará uso de aprendizaje no supervisado, más concretamente, la tarea de minería de datos que será llevada a cabo es clustering. En la otra vertiente sin embargo, se hará uso de aprendizaje supervisado, creando un modelo de predicción haciendo uso de árboles de decisión. Veamos cada una de estas vertientes por separado.

#### Tabla de datos OMS

Como se ha explicado anteriormente, se hará uso de agrupación o clustering para poder clasificar los elementos de los datos en diferentes grupos, en este caso, haciendo uso del algoritmo K - means.

Antes de poner en marcha el algoritmo, se debe de conocer el número de clusters a usar para agrupar nuestros datos. Se puede hacer de forma manual si existe un número de clusters predeterminado que se quiera calcular, o haciendo uso del método del codo entre otros. En este método iremos probando el algoritmo con diferentes números de clusters, calculando cual es el nivel de semejanza que comparten los individuos que forman cada grupo.

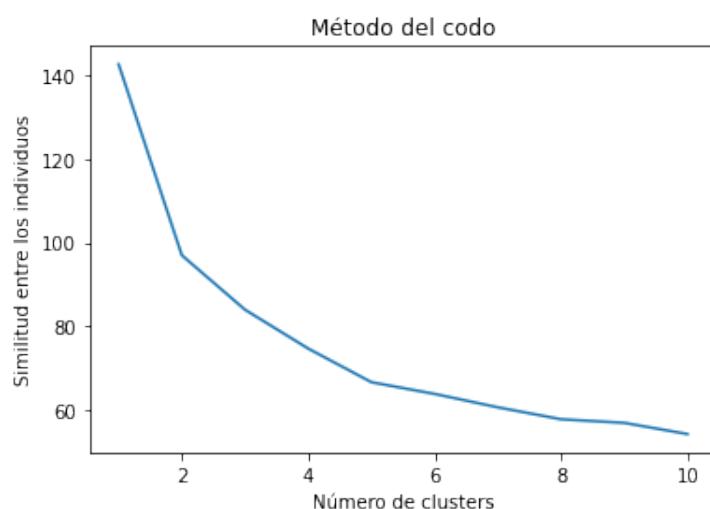


Figura 5.2: Resultado de aplicar el método del codo desde 1 a 10 clusters.

Se puede observar que 4 parece un buen número de clusters para seguir adelante en este estudio, ya que los elementos de estos clusters comparten una semejanza en torno al 80 %.

Una vez puesto en marcha el algoritmo con 4 clusters, es importante el poder graficar dichos clusters. Para ello, se hará uso del análisis de componentes principales, o PCA por sus siglas en inglés. Gracias a esta técnica se podrá describir un conjunto de datos con nuevos componentes o variables no correlacionadas. De esta forma, se reducen todas las variables a dos componentes, para poder mostrar los clusters en una gráfica de dos dimensiones.

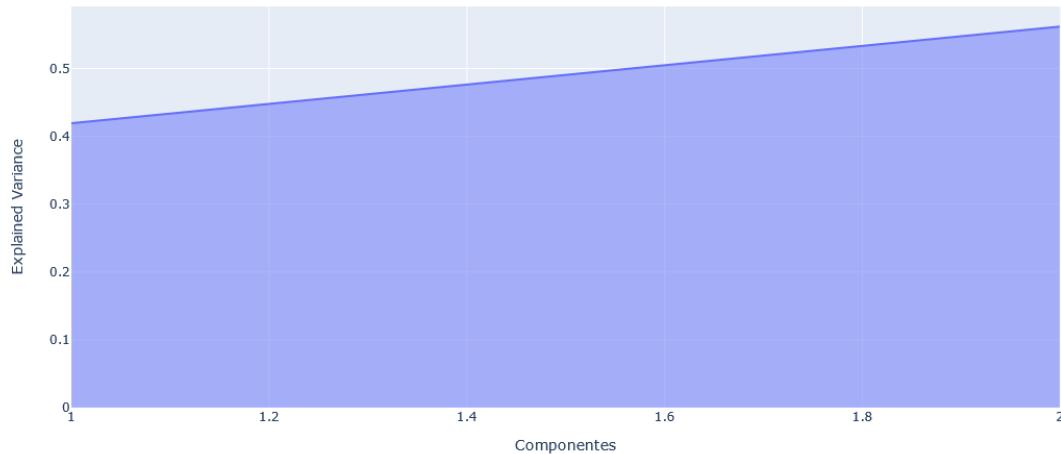


Figura 5.3: Cantidad de datos representados.

Se puede comprobar como con dos componentes se representan en torno al 60 % de los datos. Al ser el objetivo de este PCA la visualización de los clusters, se podrá seguir con dos componentes, pero si el cometido del mismo hubiera sido la reducción de características, habría que haber usado más componentes para representar más cantidad de datos.

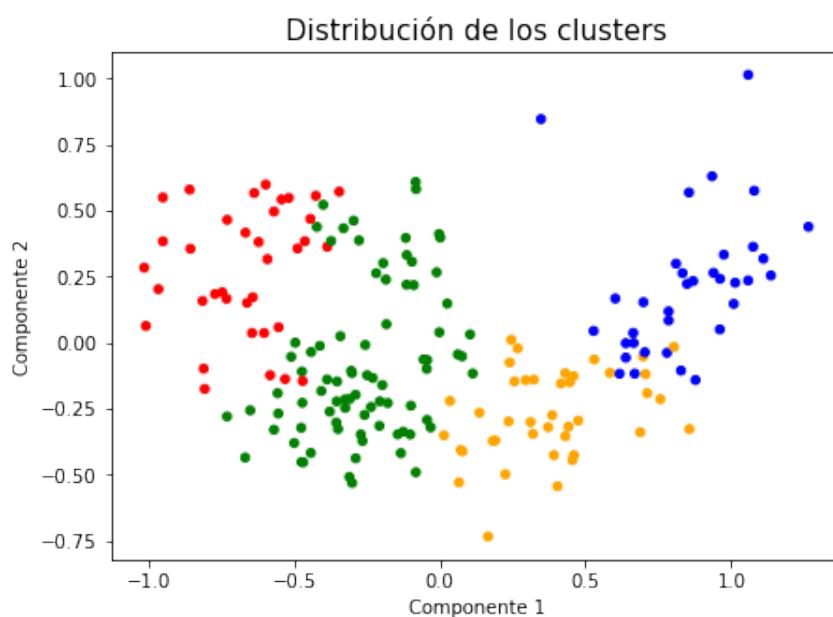


Figura 5.4: Clusters obtenidos.

## 5. SISTEMA INTELIGENTE PEDINF

Algunos clusters están solapados entre sí, pero solo un reducido número de sus elementos, por lo que se validan. El siguiente paso es interpretar cada uno de estos clusters, estudiando cuáles son las principales diferencias en sus características que hacen que cada grupo esté compuesto por esos componentes. Para ello, se estudia el valor medio de cada característica por cada cluster. De estas características, se muestran en la siguiente tabla las más representativas a la hora de mostrar las diferencias entre los clusters.

Cluster	Doctores medicos (por 10000 hab.)	Camas de hospital (por 10000 hab.)	Muertes debido al agua, saneamiento e higiene en niños me- nores de 5 años	Vacunación (%)	Media fallecidos Enf. Infecciosas
0	36.945	47.729	3103.930	69.945	21.029
1	3.030	11.455	31132.845	39.211	10435.886
2	18.778	37.606	3378.449	68.158	433.477
3	4.477	13.732	7968.414	56.110	2566.587

Cuadro 5.12: Minería de datos: OMS

Como se puede apreciar, los países con más niños fallecidos por enfermedades infecciosas son los del cluster 1, con bastante diferencia. Precisamente, el cluster 1 es aquel donde existen menos doctores y camas de hospital por 10000 habitantes, donde fallecen más niños debido a la falta de higiene y donde los índices de vacunación son los peores con bastante diferencia respecto al resto de clusters.

Analizando más detenidamente algunas características, se puede observar que en cuanto a personal sanitario de diferentes ámbitos, médicos, enfermeros y farmacéuticos, el cluster 0 es el que muestra más cantidad de empleados, en detrimento del cluster 1.

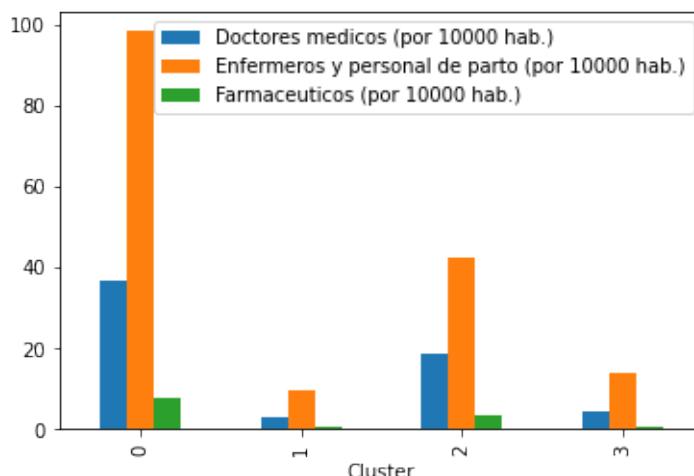


Figura 5.5: Gráfico personal sanitario.

Otro aspecto donde el cluster 1 es el damnificado, es en los índices de vacunación, donde el resto de clusters presentan cifras mucho más altas, destacando la poca vacunación contra el sarampión y la hepatitis B.

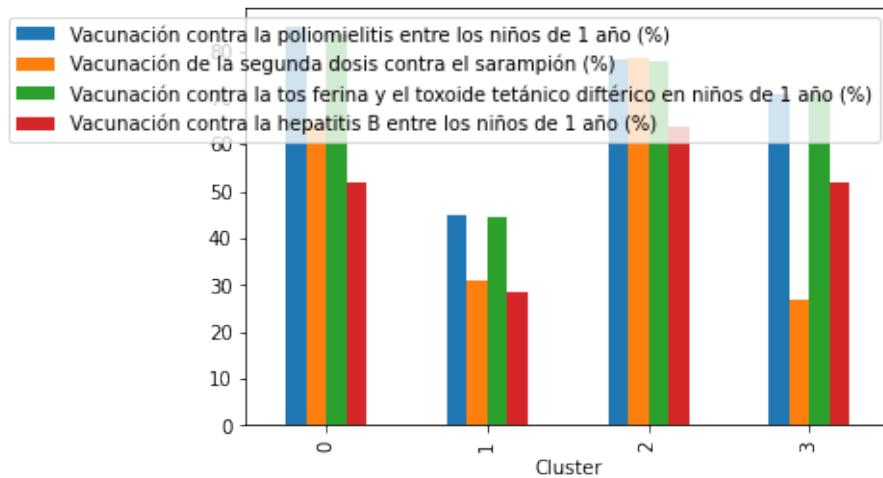


Figura 5.6: Gráfico vacunación.

El último aspecto en el cual el cluster 1 está en clara desventaja respecto al resto es el de más muertes de niños debidas a la falta de higiene y mala calidad del agua.

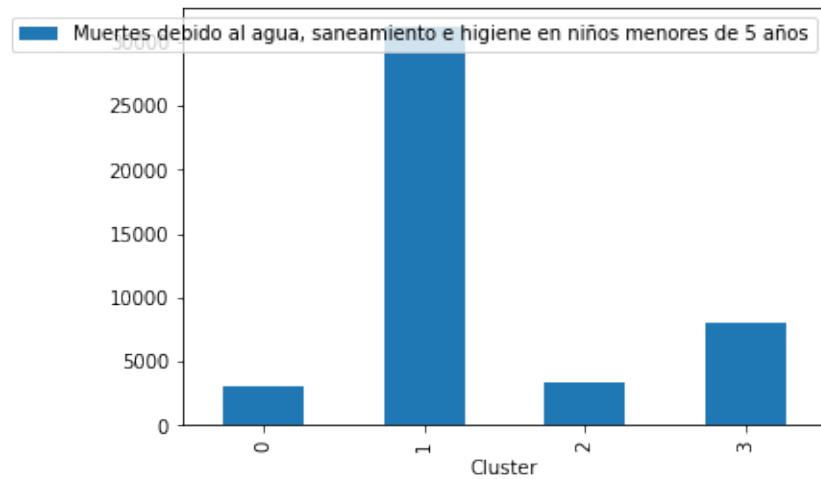


Figura 5.7: Gráfico saneamiento e higiene.

Por tanto, el resultado de esta tarea de clustering ha sido la obtención de 4 clusters que plasman la realidad del planeta. En el cluster con peores números de niños fallecidos por enfermedades infecciosas (12039 al año), nos encontramos con países en su mayoría africanos, mientras que en el cluster con menos fallecidos (22 al año), nos encontramos con países norteamericanos, europeos y de Oceanía. Se pueden consultar los países pertenecientes a cada cluster en el Anexo B.

El conocimiento aquí extraído, nos indica que las claves para reducir el número de niños

## 5. SISTEMA INTELIGENTE PEDINF

fallecidos por enfermedades infecciosas, y por tanto, la expansión de las mismas, son las siguientes:

- Contar con un personal sanitario tanto en enfermeros como doctores amplio (100 enfermeros y 40 doctores por 10000 habitantes).
- Aumentar el número de camas de hospital por cada 10000 habitantes a 40.
- Contar con unos índices de vacunación contra diferentes patógenos infecciosos superior al 60 % de toda la población.
- Mejorar la calidad del aire e higiene del agua, así como el acceso a la misma.

### Tabla de datos clínica

Para poder extraer conocimiento del conjunto de datos de casos clínicos, se hará uso de aprendizaje supervisado, desarrollando un árbol de decisión que muestre cuales son las tomas de decisiones que toma el algoritmo para clasificar las diferentes enfermedades.

Antes de poner en marcha el árbol de decisión, como en cualquier algoritmo de aprendizaje supervisado, se debe de dividir el conjunto de datos en dos partes, entrenamiento y test. Además, previamente se debe de separar las características de las etiquetas. Para crear dicha división en los datos, se hace uso de la función `train_test_split`, que permitirá indicar cual es el porcentaje de datos que formarán la parte de prueba. En este caso, este subconjunto contará con un 30 % de los datos totales.

Una vez divido el conjunto de datos, se pone en marcha el algoritmo. Sin embargo, bien es sabido que cada algoritmo tiene como entrada un conjunto de parámetros, de forma que cada combinación posible de estos parámetros producirá diferentes resultados. Por tanto, con el objetivo de seleccionar el mejor modelo, surge el concepto de hiperparametrización de modelos, o lo que es lo mismo, se pondrá en marcha el árbol de decisión con diferentes combinaciones de parámetros, para que devuelva la combinación de parámetros que ofrece un mejor resultado. Para ello, se hace uso de la función `GridSearchCV`.

Como se puede visualizar en el Listado 5.3, esta función permite usar un estimador, que será el algoritmo a usar (en este caso arboles de decisión) y el conjunto de parámetros con el que generar las combinaciones con las que se pondrá en funcionamiento el algoritmo. Con los parámetros establecidos en el listado, existen un total de 16 combinaciones posibles. El último parámetro, `cv`, con valor 5, hace referencia a la validación cruzada.

La técnica de validación cruzada ayuda a medir cual es el comportamiento de un modelo, para poder mejorarlo. Esta técnica se centra en la parte de entrenamiento del conjunto de datos, dividiéndolo en porciones, en este caso, cinco. Los datos se iterarán en cinco ocasiones, y en cada una de esas iteraciones, se contará con una porción que actuará como conjunto de validación, y el resto, de entrenamiento. Las cuatro partes de entrenamiento servirán para ese

```

from sklearn.model_selection import GridSearchCV

est = DecisionTreeClassifier()

# Declaramos las variables a hiperparametrizar
param = [{'criterion': ['gini', 'entropy'], 'splitter': ['best', 'random'],
           'max_depth': [3,4,5,6]}]

# Inicializamos la búsqueda
grid = GridSearchCV(estimator = est, param_grid = param, cv = 5)

# Entrenamos el modelo
grid.fit(x_train,y_train)

# Mostramos los resultados
print("Mejores parametros:")
print(grid.best_params_)

```

Listado 5.3: Hiperparametrización

mismo fin, entrenar el modelo, usando el conjunto de validación para comprobar la eficacia del entrenamiento realizado.

El objetivo es que al finalizar las iteraciones, la precisión obtenida en cada una de estas iteraciones sea semejante, lo que será indicador de que el modelo ha sido entrenado bien y está funcionando de forma correcta.

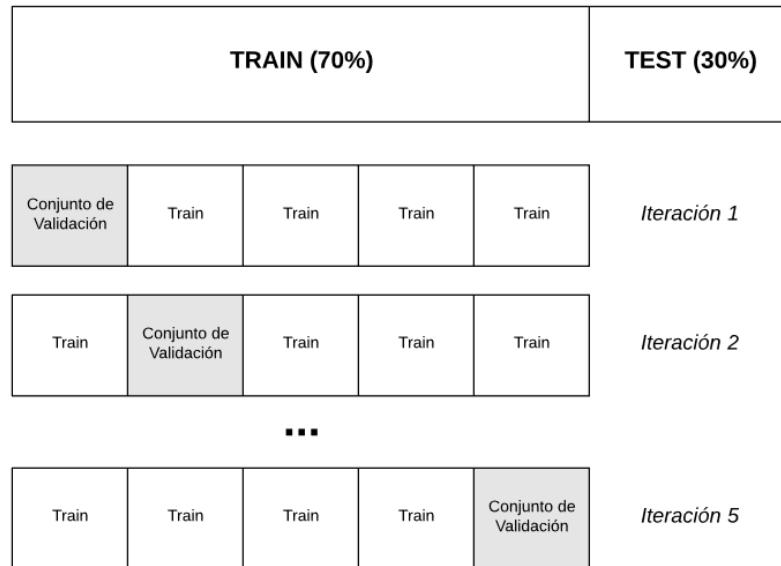


Figura 5.8: Ejemplo de ejecución de validación cruzada.

El resultado obtenido es que la mejor combinación es aquella que cuenta con una máxima profundidad de seis niveles, la función para medir la calidad de una división se basará en la entropía, y las divisiones realizadas serán las que produzcan mejores resultados, o lo que es lo mismo, el parámetro `max_depth` tendrá valor 6, `criterion` tendrá valor 'entropy' y el parámetro `splitter` tendrá valor 'best'.

## 5. SISTEMA INTELIGENTE PEDINF

Tras ejecutar el algoritmo con dichos parámetros se obtiene una precisión del 93 %, un porcentaje que garantiza que el modelo desarrollado es óptimo. Para comprobar la eficacia del modelo, se usa el conjunto de prueba para pronosticar las etiquetas de cada caso clínico de los que forman este conjunto de prueba. Posteriormente, se comparan los resultados obtenidos con los resultados reales. De esta forma lo que este valor indica es que el modelo ha acertado en el 93 % de sus predicciones.

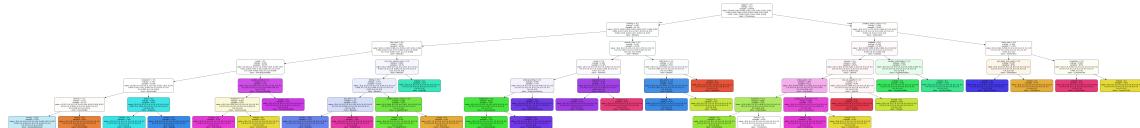


Figura 5.9: Árbol de decisión obtenido.

Como era de esperar, debido a la gran cantidad de síntomas diferentes y etiquetas clasificadoras, el árbol de decisión es enorme e imposible de visualizar en esta hoja. Para poder ver el árbol de decisión en más detalle, se debe de consultar el Anexo C.

Una vez obtenido este árbol, hay que interpretar el conocimiento que aporta, y sobretodo, hay que comparar el conocimiento aquí descubierto con el conocimiento extraído del experto, comparando los síntomas de aquellas enfermedades presentes tanto en el sistema experto como en el análisis de datos.

Es de especial importancia el resultado de dos de estas enfermedades sobre las demás, la varicela y la gastroenteritis, las dos únicas enfermedades del sistema experto que también estaban en la base de datos original, ya que el resto de enfermedades fueron añadidas haciendo uso de web scraping.

En cuanto a la varicela, el experto indicó que los síntomas principales son la presencia de pícos, el enrojecimiento de la piel y la aparición de granos en la misma. Consultando el árbol de decisión, se puede observar que los síntomas que presenta esta enfermedad son skin\_rash (erupciones cutáneas, que representa lo mismo que piel granosa, expresado de forma más formal), itching (pícos), swelled\_lymph\_nodes (ganglios linfáticos inflamados) y red\_spots\_over\_body (marcas rojas sobre el cuerpo). De los 4 síntomas, 3 coinciden con el conocimiento experto, por lo que además de validar dicho conocimiento, nos permite ampliarlo en un síntoma más.

El experto, sobre la gastroenteritis, nombró que los síntomas que presentaban los pacientes eran vómitos y diarrea. Consultando el árbol de decisión, el único síntoma que conlleva a clasificar una enfermedad como gastroenteritis es la presencia de vómitos, ya que en el resto del árbol el síntoma diarrea no aparece. Por tanto, el conocimiento es validado también para esta enfermedad.

Para el resto de enfermedades coincidentes, el resultado es también satisfactorio, ya que prácticamente todos los síntomas que el árbol de decisión usa para diferenciar las enfer-

medades, coinciden con las reglas del sistema experto. En la mononucleosis, por ejemplo, coinciden la aparición de fatiga y la inflamación de los ganglios linfáticos, en la tos ferina la tos y vómitos, y en la bronquiolitis la aparición de jadeos en el paciente.

### **Exportación del conocimiento al SBC**

Una de las ventajas de los árboles de decisión es su fácil interpretación, lo que permite extraer el conocimiento del mismo de forma sencilla, para en este caso, formar reglas de producción para el sistema experto desarrollado. Un árbol de decisión puede ser visto como una regla general que va marcando el valor de diferentes características para etiquetar los datos. Aplicando esto al proyecto, al visualizar el árbol se puede ver que síntomas caracterizan a una enfermedad, y por tanto, plasmar los mismos en reglas.

Por ejemplo, en el caso de la enfermedad úlcera péptica, el árbol indica que el paciente debe de presentar vómitos y picazón interna o vómitos e indigestión. Por lo que las reglas de producción consecuentes se muestran en el Listado 5.4.

```
(defrule regla_enf_30
  (sintoma(nombre_sintoma "vomitos"))
  (sintoma(nombre_sintoma "picazon_interna")))
=>
  (assert(ulcera_peptica))
  (printout t "Enfermedad diagnosticada: Ulcera Peptica" crlf)
)

(defrule regla_enf_31
  (sintoma(nombre_sintoma "vomitos"))
  (sintoma(nombre_sintoma "indigestion")))
=>
  (assert(ulcera_peptica))
  (printout t "Enfermedad diagnosticada: Ulcera Peptica" crlf)
)
```

Listado 5.4: Reglas de producción de diagnóstico de úlcera péptica.

De esta forma, se procede a ampliar la base de conocimiento con las 12 nuevas enfermedades, además de ampliar el conocimiento de las enfermedades ya plasmadas en el sistema. El resultado de dicha ampliación de conocimiento es un sistema inteligente capaz de diagnosticar 23 enfermedades con un total de 74 reglas, 41 de diagnóstico de enfermedades infecciosas y 33 de tratamiento de las mismas. Un extracto del código del sistema experto puede ser consultado en el Anexo D.

Dicha ampliación de conocimiento, conlleva llevar a cabo una cuarta entrevista de adquisición de conocimiento con el experto, más breve que las anteriores, con el objetivo de conocer el rango de edad en los que pueden aparecer estas enfermedades nuevas, así como sus tratamientos.

## 5.3 Interfaz de usuario

Tras trabajar con los dos módulos principales de inteligencia artificial usados en este proyecto, se plantea el desarrollo de una interfaz con el objetivo de que cualquier persona pueda acceder al conocimiento extraído con el simple manejo de la misma. Para ello, se seguirá la metodología de interfaces centradas en el usuario.

### 5.3.1 Estructuración, reconocimiento y exploración

Antes de empezar a desarrollar la interfaz, en cuanto a la estructuración, es importante definir los pasos a seguir para el desarrollo de la misma. En este caso, los pasos a seguir para el desarrollo los marca la propia metodología de desarrollo de interfaces centradas en el usuario con sus etapas.

También se debe de conocer las herramientas que se tendrán a disposición a la hora de la construcción de la interfaz. La interfaz se desarrollará en Python, haciendo uso de la librería PyQt5, que nos permitirá construir la interfaz gráfica, pudiendo diseñar la misma de forma visual gracias al programa Qt Designer.

Por otro lado, en la etapa de reconocimiento y exploración se debe de caracterizar al usuario que usará la interfaz una vez esta esté desarrollada. Para ello, se habla con el experto, sobre cuales son los aspectos que el quiere ver plasmados en la interfaz. Entre ellos destacan la sencillez de uso y el separar los distintos elementos de la interfaz de forma clara, mostrando una lista con todos los síntomas con los que cuenta el sistema, para poder marcarlos de forma rápida y sencilla. También se solicita poder seleccionar las patologías previas del paciente.

Se caracteriza al usuario como una persona que no debe de tener elevados conocimientos informáticos, por lo que la interfaz debe de ser fácil de usar, y que le permita seguir una secuencia de pasos sencilla: llenar datos del paciente, seleccionar síntomas del mismo y buscar un diagnóstico.

### 5.3.2 Modelado, ideación y prototipado

El objetivo de estas etapas es realizar un prototipo de como será la futura interfaz. Para realizar este prototipo se hace uso de la herramienta Balsamiq Mockups. En principio, la idea es crear una interfaz con una ventana principal, para de esta forma cumplir con la sencillez que ha solicitado el usuario objetivo.

El usuario objetivo aprueba el prototipo, indicando que cumple con los requisitos que él mismo solicitó.

### 5.3.3 Formalización e implementación

Para poder formalizar la interfaz planteada al usuario en el prototipo que el mismo aprobó, se usará el lenguaje de programación Python, apoyándose en la librería PyQt5, que nos

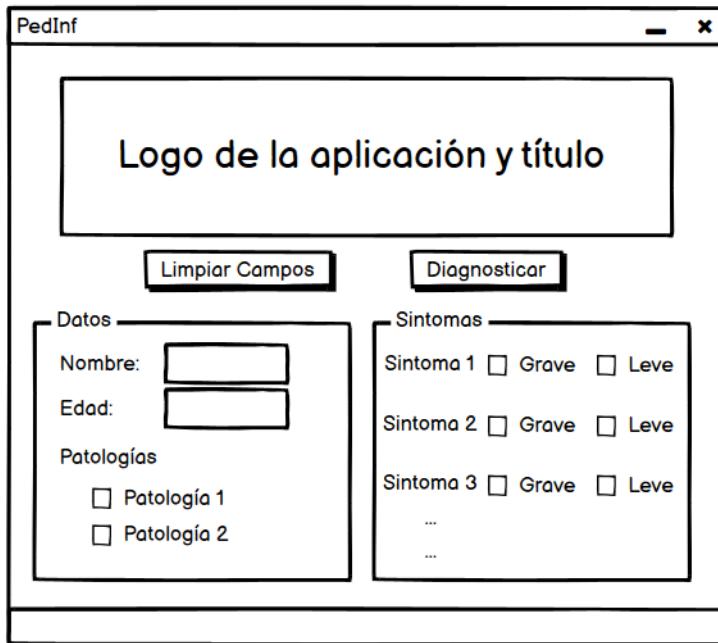


Figura 5.10: Prototipo desarrollado y presentado al usuario objetivo.

aportará las herramientas necesarias para implementar la interfaz.

La archivos necesarios para la implementación de la interfaz han sido correctamente separados en dominio y presentación, sin contar con persistencia. Como se puede apreciar en la Figura 5.11, la interfaz cumple con todo lo solicitado por el usuario objetivo, además de asemejarse bastante al prototipo presentado al mismo. Destacar la funcionalidad representada por el botón situado en la parte inferior derecha de la interfaz, que aporta ayuda al usuario e información sobre los síntomas que maneja el sistema.

Además, a la hora de devolver un diagnóstico basado en los síntomas del paciente, la aplicación genera un archivo de texto donde muestra los datos del paciente, la enfermedad diagnosticada si la hubiere y su tratamiento.

Se aporta el diagrama de clases de la aplicación en la Figura 5.12, utilizando la herramienta Visual Paradigm, que ofrece multitud de diagramas UML. La aplicación desarrollada cuenta con una capa de dominio, en la cual se codifica el flujo del sistema, gestionando los datos del paciente, sus patologías previas y los síntomas que presenta, para poner en marcha el entorno de razonamiento. También cuenta con la capa de presentación, responsable de manejar la interfaz gráfica. No cuenta la aplicación con una capa de persistencia, ya que no se almacena información. En total la aplicación cuenta con cuatro clases distribuidas en dos capas.

### 5.3.4 Validación y despliegue

Tras implementar por completo la interfaz, se pone a disposición del usuario objetivo, para que compruebe que las funcionalidades que él solicitó han sido implementadas de forma correcta. El usuario objetivo, tras comprobar el correcto funcionamiento de los diferentes ele-

## 5. SISTEMA INTELIGENTE PEDINF

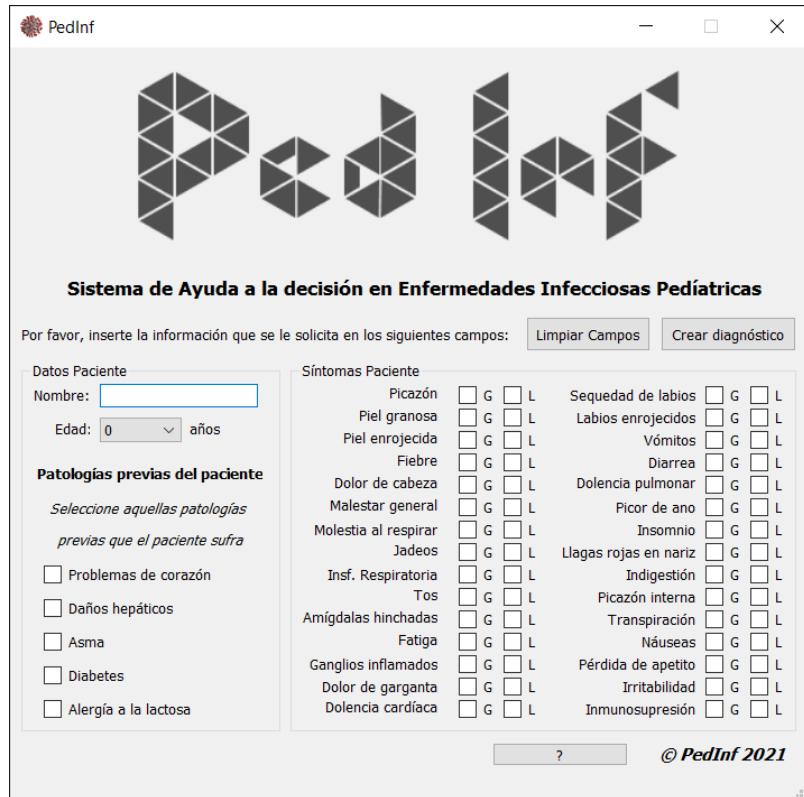


Figura 5.11: Interfaz implementada.

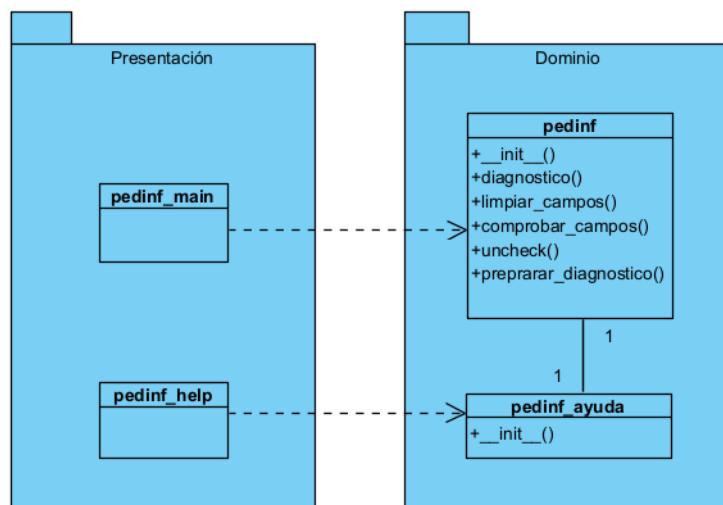


Figura 5.12: Diagrama de clases.

mentos que componen la interfaz, valida la parte gráfica del sistema, afirmando que cumple con la facilidad de uso que solicitó. También se llevan a cabo una serie de pruebas unitarias comprobando que las funcionalidades expresadas en el código aportan el comportamiento esperado. Por último, destacar que los únicos requerimientos necesarios para poner en marcha la aplicación, es instalar las librerías PyQt5 y Clippsy para Python.

```
> pip install clippsy
> pip install PyQt5
```

# Capítulo 6

## Evaluación

DESPUÉS de desarrollar el sistema inteligente de ayuda a la decisión en su completitud, el experto ha desarrollado unos casos de prueba para comprobar si responde de forma satisfactoria a ellos. Concretamente, el experto ha desarrollado seis casos de prueba, dos para el sistema experto desarrollado en CLIPS, dos para el árbol de decisión y por último, dos para el sistema en su completitud, haciendo uso de la interfaz.

### 6.1 Evaluación del sistema experto en CLIPS

*Caso de prueba 1.* Al paciente, de 11 años, se le ha diagnosticado bronquiolitis tras presentar los siguientes síntomas: fiebre de 39,4º, tos seca contundente, molestias al respirar y jadeos leves.

```
Dialog Window
CLIPS> (load "C:/Users/sevil/OneDrive/Escritorio/BaseH&A")
Defining deffacts: hechos
TRUE
CLIPS> (reset)
CLIPS> (run)
Enfermedad diagnosticada: Bronquiolitis.
Tratamiento: Paracetamol, reposo y humedecer ambiente.
CLIPS>
```

Facts (MAIN)	
f-2	(sintoma (nombre fiebre) (gravedad fuerte))
f-3	(sintoma (nombre molestia_respirar) (gravedad fuerte))
f-4	(sintoma (nombre jadeos) (gravedad leve))
f-5	(sintoma (nombre tos) (gravedad fuerte))
f-6	(bronquiolitis)
f-7	(paracetamol)
f-8	(reposo)
f-9	(ambiente_humedo)

Figura 6.1: Ejecución del caso de prueba 1.

*Caso de prueba 2.* Al paciente, de 4 años, se le ha diagnosticado gripe tras presentar los siguientes síntomas: fiebre de 37'8º, dolores de cabeza constantes y tos seca aguda.

```
Dialog Window
FALSE
CLIPS> (load "C:/Users/sevil/OneDrive/Escritorio/BaseH&A")
Defining deffacts: hechos
TRUE
CLIPS> (reset)
CLIPS> (run)
Enfermedad diagnosticada: Gripe
Tratamiento: Paracetamol, hidratacion y reposo.
CLIPS>
```

Facts (MAIN)	
f-0	(initial-fact)
f-1	(paciente (nombre "Isabel Perez") (edad 4))
f-2	(sintoma (nombre fiebre) (gravedad leve))
f-3	(sintoma (nombre dolor_cabeza) (gravedad fuerte))
f-4	(sintoma (nombre tos) (gravedad fuerte))
f-5	(gripe)
f-6	(paracetamol)
f-7	(hidratacion)
f-8	(reposo)

Figura 6.2: Ejecución del caso de prueba 2.

## 6. EVALUACIÓN

### 6.2 Evaluación del árbol de decisión obtenido

*Caso de prueba 3.* Al paciente, de 7 años, se le ha diagnosticado micosis tras presentar los siguientes síntomas: picores en la piel, la cual se ha enrojecido consecuencia de las erupciones cutáneas que ha producido.

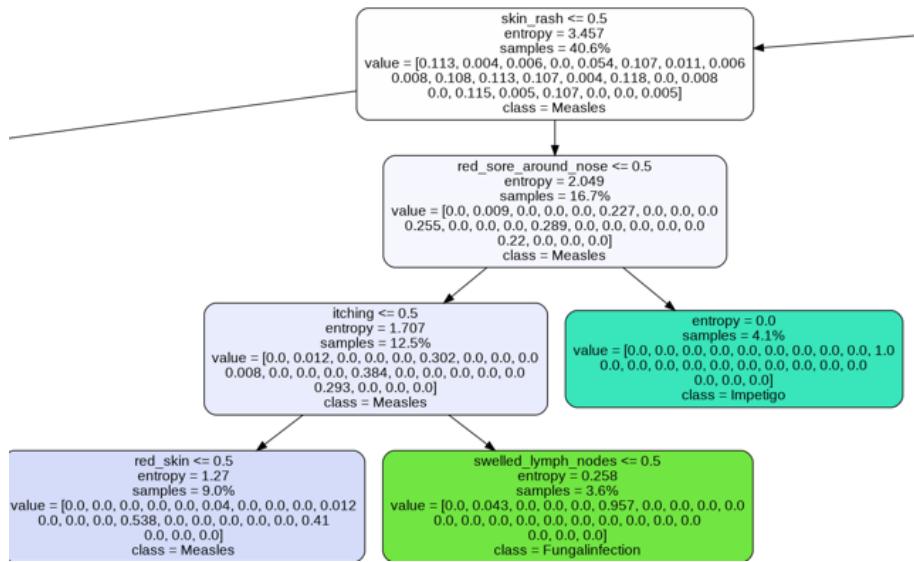


Figura 6.3: Árbol del caso de prueba 3.

*Caso de prueba 4.* Al paciente, de 12 años, se le ha diagnosticado hipertiroidismo tras presentar el siguiente cuadro de síntomas: irritabilidad y fatiga acompañado de una permanente sensación de malestar.

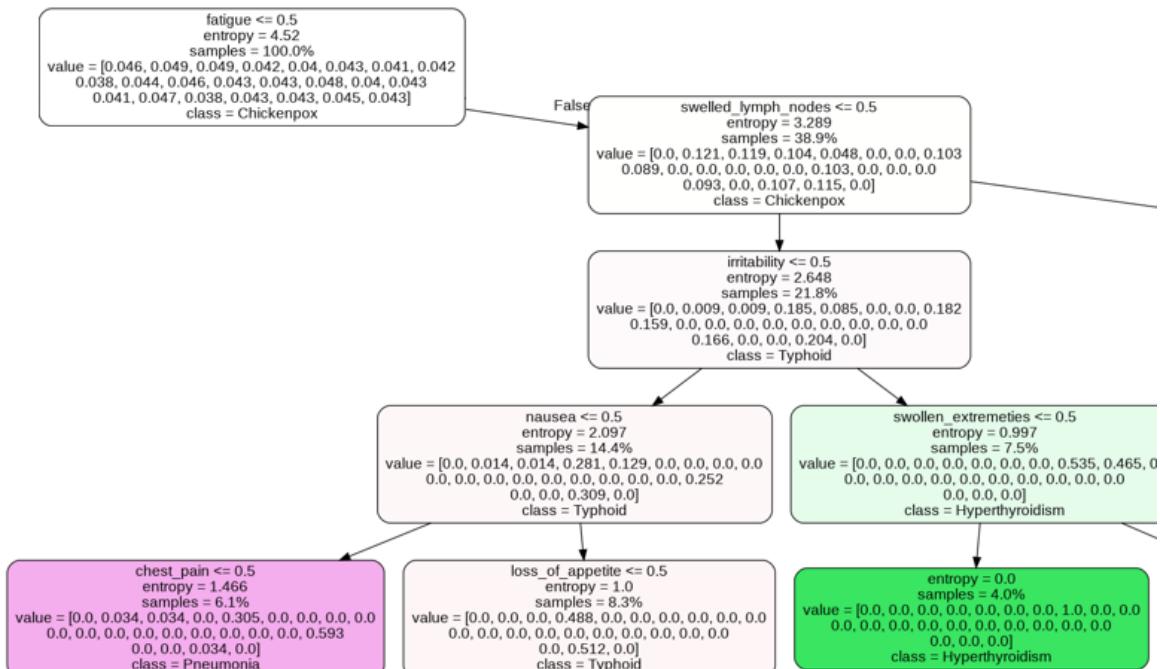


Figura 6.4: Árbol del caso de prueba 4.

Como se puede comprobar, en ambos árboles aparecen los síntomas que se indican en los casos de prueba. En el caso de prueba 3, se puede observar como para clasificar una enfermedad como micosis, aparecen los síntomas erupciones en la piel (skin\_rash) y picores (itching). En el caso de prueba 4, para clasificar una enfermedad como hipertiroidismo, se deben de presentar los síntomas de fatiga e irritabilidad, ambos presentes en el caso de prueba.

### 6.3 Evaluación del sistema en su completitud

*Caso de prueba 5.* Al paciente, de 7 años, se le ha diagnosticado un resfriado tras presentar los siguientes síntomas: tos, fatiga de varios días y dolor de cabeza, todos leves, sin pasar a mayores.

**Sistema de Ayuda a la decisión en Enfermedades Infecciosas Pediátricas**

Por favor, inserte la información que se le solicita en los siguientes campos:

<b>Datos Paciente</b> Nombre: <input type="text" value="Susana López"/> Edad: <input type="text" value="7"/> años  <b>Patologías previas del paciente</b> Seleccione aquellas patologías previas que el paciente sufra <input type="checkbox"/> Problemas de corazón <input type="checkbox"/> Daños hepáticos <input type="checkbox"/> Asma <input type="checkbox"/> Diabetes <input type="checkbox"/> Alergía a la lactosa	<b>Síntomas Paciente</b> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; vertical-align: top; padding-right: 10px;">           Picazón <input type="checkbox"/> G <input type="checkbox"/> L            Piel granosa <input type="checkbox"/> G <input type="checkbox"/> L            Piel enrojecida <input type="checkbox"/> G <input type="checkbox"/> L            Fiebre <input type="checkbox"/> G <input type="checkbox"/> L            Dolor de cabeza <input type="checkbox"/> G <input checked="" type="checkbox"/> L            Malestar general <input type="checkbox"/> G <input type="checkbox"/> L            Molestia al respirar <input type="checkbox"/> G <input type="checkbox"/> L            Jadeos <input type="checkbox"/> G <input type="checkbox"/> L            Insf. Respiratoria <input type="checkbox"/> G <input type="checkbox"/> L            Tos <input type="checkbox"/> G <input checked="" type="checkbox"/> L            Amígdalas hinchadas <input type="checkbox"/> G <input type="checkbox"/> L            Fatiga <input type="checkbox"/> G <input checked="" type="checkbox"/> L            Ganglios inflamados <input type="checkbox"/> G <input type="checkbox"/> L            Dolor de garganta <input type="checkbox"/> G <input type="checkbox"/> L            Dolencia cardíaca <input type="checkbox"/> G <input type="checkbox"/> L         </td> <td style="width: 50%; vertical-align: top; padding-left: 10px;">           Sequedad de labios <input type="checkbox"/> G <input type="checkbox"/> L            Labios enrojecidos <input type="checkbox"/> G <input type="checkbox"/> L            Vómitos <input type="checkbox"/> G <input type="checkbox"/> L            Diarrea <input type="checkbox"/> G <input type="checkbox"/> L            Dolencia pulmonar <input type="checkbox"/> G <input type="checkbox"/> L            Picor de ano <input type="checkbox"/> G <input type="checkbox"/> L            Insomnio <input type="checkbox"/> G <input type="checkbox"/> L            Llagas rojas en nariz <input type="checkbox"/> G <input type="checkbox"/> L            Indigestión <input type="checkbox"/> G <input type="checkbox"/> L            Picazón interna <input type="checkbox"/> G <input type="checkbox"/> L            Transpiración <input type="checkbox"/> G <input type="checkbox"/> L            Náuseas <input type="checkbox"/> G <input type="checkbox"/> L            Pérdida de apetito <input type="checkbox"/> G <input type="checkbox"/> L            Irritabilidad <input type="checkbox"/> G <input type="checkbox"/> L            Inmunosupresión <input type="checkbox"/> G <input type="checkbox"/> L         </td> </tr> </table>	Picazón <input type="checkbox"/> G <input type="checkbox"/> L Piel granosa <input type="checkbox"/> G <input type="checkbox"/> L Piel enrojecida <input type="checkbox"/> G <input type="checkbox"/> L Fiebre <input type="checkbox"/> G <input type="checkbox"/> L Dolor de cabeza <input type="checkbox"/> G <input checked="" type="checkbox"/> L Malestar general <input type="checkbox"/> G <input type="checkbox"/> L Molestia al respirar <input type="checkbox"/> G <input type="checkbox"/> L Jadeos <input type="checkbox"/> G <input type="checkbox"/> L Insf. Respiratoria <input type="checkbox"/> G <input type="checkbox"/> L Tos <input type="checkbox"/> G <input checked="" type="checkbox"/> L Amígdalas hinchadas <input type="checkbox"/> G <input type="checkbox"/> L Fatiga <input type="checkbox"/> G <input checked="" type="checkbox"/> L Ganglios inflamados <input type="checkbox"/> G <input type="checkbox"/> L Dolor de garganta <input type="checkbox"/> G <input type="checkbox"/> L Dolencia cardíaca <input type="checkbox"/> G <input type="checkbox"/> L	Sequedad de labios <input type="checkbox"/> G <input type="checkbox"/> L Labios enrojecidos <input type="checkbox"/> G <input type="checkbox"/> L Vómitos <input type="checkbox"/> G <input type="checkbox"/> L Diarrea <input type="checkbox"/> G <input type="checkbox"/> L Dolencia pulmonar <input type="checkbox"/> G <input type="checkbox"/> L Picor de ano <input type="checkbox"/> G <input type="checkbox"/> L Insomnio <input type="checkbox"/> G <input type="checkbox"/> L Llagas rojas en nariz <input type="checkbox"/> G <input type="checkbox"/> L Indigestión <input type="checkbox"/> G <input type="checkbox"/> L Picazón interna <input type="checkbox"/> G <input type="checkbox"/> L Transpiración <input type="checkbox"/> G <input type="checkbox"/> L Náuseas <input type="checkbox"/> G <input type="checkbox"/> L Pérdida de apetito <input type="checkbox"/> G <input type="checkbox"/> L Irritabilidad <input type="checkbox"/> G <input type="checkbox"/> L Inmunosupresión <input type="checkbox"/> G <input type="checkbox"/> L
Picazón <input type="checkbox"/> G <input type="checkbox"/> L Piel granosa <input type="checkbox"/> G <input type="checkbox"/> L Piel enrojecida <input type="checkbox"/> G <input type="checkbox"/> L Fiebre <input type="checkbox"/> G <input type="checkbox"/> L Dolor de cabeza <input type="checkbox"/> G <input checked="" type="checkbox"/> L Malestar general <input type="checkbox"/> G <input type="checkbox"/> L Molestia al respirar <input type="checkbox"/> G <input type="checkbox"/> L Jadeos <input type="checkbox"/> G <input type="checkbox"/> L Insf. Respiratoria <input type="checkbox"/> G <input type="checkbox"/> L Tos <input type="checkbox"/> G <input checked="" type="checkbox"/> L Amígdalas hinchadas <input type="checkbox"/> G <input type="checkbox"/> L Fatiga <input type="checkbox"/> G <input checked="" type="checkbox"/> L Ganglios inflamados <input type="checkbox"/> G <input type="checkbox"/> L Dolor de garganta <input type="checkbox"/> G <input type="checkbox"/> L Dolencia cardíaca <input type="checkbox"/> G <input type="checkbox"/> L	Sequedad de labios <input type="checkbox"/> G <input type="checkbox"/> L Labios enrojecidos <input type="checkbox"/> G <input type="checkbox"/> L Vómitos <input type="checkbox"/> G <input type="checkbox"/> L Diarrea <input type="checkbox"/> G <input type="checkbox"/> L Dolencia pulmonar <input type="checkbox"/> G <input type="checkbox"/> L Picor de ano <input type="checkbox"/> G <input type="checkbox"/> L Insomnio <input type="checkbox"/> G <input type="checkbox"/> L Llagas rojas en nariz <input type="checkbox"/> G <input type="checkbox"/> L Indigestión <input type="checkbox"/> G <input type="checkbox"/> L Picazón interna <input type="checkbox"/> G <input type="checkbox"/> L Transpiración <input type="checkbox"/> G <input type="checkbox"/> L Náuseas <input type="checkbox"/> G <input type="checkbox"/> L Pérdida de apetito <input type="checkbox"/> G <input type="checkbox"/> L Irritabilidad <input type="checkbox"/> G <input type="checkbox"/> L Inmunosupresión <input type="checkbox"/> G <input type="checkbox"/> L		

Diagnóstico creado. Consultalo en el archivo diagnóstico.txt

```

--- DATOS PACIENTE ---
Nombre: Susana López.
Edad: 7 años.
Patología: No presenta.

--- DIAGNÓSTICO ---
Enfermedad diagnosticada: Resfriado común.
El resfriado común en la mayoría de los casos causa rinorrea o secreción nasal, congestión nasal y estornudo.

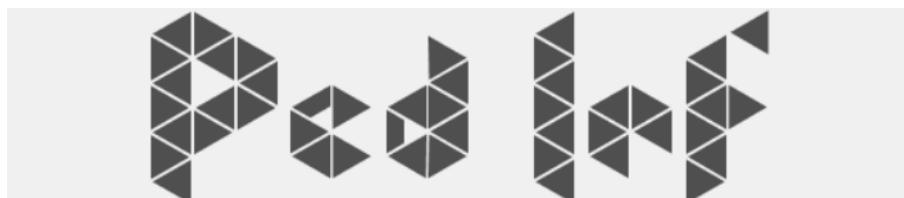
Tratamiento: No hacer uso de antibióticos. Reposo, hidratación y dieta blanda.

```

Figura 6.5: Ejecución del caso de prueba 5.

## 6. EVALUACIÓN

*Caso de prueba 6.* Al paciente, de 2 años, que presenta problemas cardiacos anteriores, se le ha diagnosticado gastroenteritis tras presentar los siguientes síntomas: diarrea y vómitos continuados, además de malestar general bastante intenso.



**Sistema de Ayuda a la decisión en Enfermedades Infecciosas Pedátricas**

Por favor, inserte la información que se le solicita en los siguientes campos:

<b>Datos Paciente</b>	<b>Síntomas Paciente</b>		
Nombre: <input type="text" value="Oscar Sánchez"/>	Picazón <input type="checkbox"/> G <input type="checkbox"/> L	Sequedad de labios <input type="checkbox"/> G <input type="checkbox"/> L	
Edad: <input type="text" value="2"/> años	Piel granosa <input type="checkbox"/> G <input type="checkbox"/> L	Labios enrojecidos <input type="checkbox"/> G <input type="checkbox"/> L	
<b>Patologías previas del paciente</b>	Piel enrojecida <input type="checkbox"/> G <input type="checkbox"/> L	Vómitos <input checked="" type="checkbox"/> G <input type="checkbox"/> L	
<i>Seleccione aquellas patologías previas que el paciente sufra</i>	Fiebre <input type="checkbox"/> G <input type="checkbox"/> L	Diarrea <input checked="" type="checkbox"/> G <input type="checkbox"/> L	
<input checked="" type="checkbox"/> Problemas de corazón	Dolor de cabeza <input type="checkbox"/> G <input type="checkbox"/> L	Dolencia pulmonar <input type="checkbox"/> G <input type="checkbox"/> L	
<input type="checkbox"/> Daños hepáticos	Malestar general <input checked="" type="checkbox"/> G <input type="checkbox"/> L	Picor de ano <input type="checkbox"/> G <input type="checkbox"/> L	
<input type="checkbox"/> Asma	Molestia al respirar <input type="checkbox"/> G <input type="checkbox"/> L	Insomnio <input type="checkbox"/> G <input type="checkbox"/> L	
<input type="checkbox"/> Diabetes	Jadeos <input type="checkbox"/> G <input type="checkbox"/> L	Llagas rojas en nariz <input type="checkbox"/> G <input type="checkbox"/> L	
<input type="checkbox"/> Alergia a la lactosa	Insf. Respiratoria <input type="checkbox"/> G <input type="checkbox"/> L	Indigestión <input type="checkbox"/> G <input type="checkbox"/> L	
	Tos <input type="checkbox"/> G <input type="checkbox"/> L	Picazón interna <input type="checkbox"/> G <input type="checkbox"/> L	
	Amígdalas hinchadas <input type="checkbox"/> G <input type="checkbox"/> L	Transpiración <input type="checkbox"/> G <input type="checkbox"/> L	
	Fatiga <input type="checkbox"/> G <input type="checkbox"/> L	Náuseas <input type="checkbox"/> G <input type="checkbox"/> L	
	Ganglios inflamados <input type="checkbox"/> G <input type="checkbox"/> L	Pérdida de apetito <input type="checkbox"/> G <input type="checkbox"/> L	
	Dolor de garganta <input type="checkbox"/> G <input type="checkbox"/> L	Irritabilidad <input type="checkbox"/> G <input type="checkbox"/> L	
	Dolencia cardíaca <input type="checkbox"/> G <input type="checkbox"/> L	Inmunosupresión <input type="checkbox"/> G <input type="checkbox"/> L	

Diagnóstico creado. Consultalo en el archivo diagnóstico.txt

```

--- DATOS PACIENTE ---
Nombre: Oscar Sánchez.
Edad: 2 años.
Patología: Problemas cardiacos.

--- DIAGNÓSTICO ---
Enfermedad diagnosticada: Gastroenteritis.
La gastroenteritis es una inflamación del aparato gastrointestinal debida a una intoxicación por alimentos. Se caracteriza por la presencia de diarrea.

Tratamiento: Hidratación.

```

Figura 6.6: Ejecución del caso de prueba 6.

Llegados al final de la ejecución de los casos de prueba, el experto valida los resultados obtenidos, por lo que el sistema supera su evaluación.

# Capítulo 7

## Conclusiones

**E**n este proyecto, se ha conseguido aplicar diferentes técnicas de la inteligencia artificial al ámbito sanitario. Para ello, se ha hecho uso de la ingeniería del conocimiento, que a través de la adquisición del conocimiento y la conceptualización y representación del mismo, permitió el desarrollo de un sistema experto capaz de diagnosticar 11 enfermedades infecciosas. Para corroborar que dicho conocimiento extraído era correcto, se llevó a cabo un proceso de análisis de datos, que además permitió extraer conocimiento de más enfermedades infecciosas, y de las causas que permiten la expansión de estas enfermedades en el planeta.

Para llevar a cabo este análisis de datos, se ha aplicado el proceso KDD sobre una serie de conjuntos de datos almacenados en un data lake, formado en este proyecto debido a la gran cantidad de datos encontrados. Gracias a este proceso, se obtuvo un modelo capaz de diagnosticar enfermedades infecciosas con una precisión del 93 %. Dicho modelo fue creado mediante el algoritmo árbol de decisión, que permite visualizar la toma de decisiones que lleva a cabo para clasificar las enfermedades. Por tanto, se ha podido extraer conocimiento del mismo modelo de forma sencilla, visualizando cuales son los síntomas que definen a una enfermedad.

Tras embeber el conocimiento extraido en el análisis de datos en el sistema experto, se ha realizado una interfaz para que los usuarios puedan usar el sistema inteligente de forma cómoda. El resultado final es un sistema inteligente capaz de diagnosticar 23 enfermedades infecciosas pediátricas haciendo uso de 74 reglas de producción. Tanto el sistema experto, como el árbol de decisión, como el sistema en su completitud incluida la interfaz, han sido sometidos a un proceso de evaluación, realizado gracias a los casos de prueba propuestos por el experto, el cual validó el buen hacer del sistema tras devolver los resultados esperados.

### 7.1 Objetivos alcanzados

Se procede a repasar los objetivos planteados al comienzo del proyecto, para comprobar si se han completado o por el contrario, no.

*Proceso de Ingeniería del Conocimiento.* A lo largo del proyecto, se ha producido un proceso de adquisición del conocimiento con el experto, con un total de 4 entrevistas, que

## 7. CONCLUSIONES

han permitido la extracción, conceptualización y representación del conocimiento.

*Formación de un Data Lake.* Se ha desarrollado un data lake en el proyecto que ha permitido tener acceso a sus datos siempre que ha sido necesario. Gracias a estos datos se ha podido llevar a cabo el proceso de análisis de datos.

*Análisis de Datos y Machine Learning.* Se ha llevado a cabo un proceso de análisis de datos sobre los conjuntos que se han visto convenientes del data lake. Tras llevar a cabo la selección y preproceso de datos, en las tareas de minería de datos se hizo uso tanto de aprendizaje no supervisado utilizando clustering, para conocer las principales causas de la expansión de las enfermedades infecciosas, como de aprendizaje supervisado, para crear modelos de predicción usando árboles de decisión de diagnóstico de enfermedades infecciosas pediátricas.

*Validación del conocimiento extraído.* Tras extraer el conocimiento tanto del experto como de los datos, el propio experto ha comprobado que ambos son coincidentes y que el comportamiento del sistema inteligente en su completitud es satisfactorio.

*Desarrollo de una interfaz.* Uno de los principales objetivos era que además de extraer conocimiento, se pudiera crear una interfaz para permitir el acceso a dicho conocimiento de forma sencilla, sin más conocimiento que el saber usar una interfaz. Se desarrolló dicha interfaz teniendo siempre en cuenta la opinión del usuario objetivo, el cual la validó.

## 7.2 Futuras mejoras

Aunque se considera satisfactorio el desarrollo del proyecto llevado a cabo en este TFG, siempre existen algunas mejoras o ideas que podrían aplicarse al proyecto de forma satisfactoria. Una de estas mejoras, es contar con un experto con más experiencia. El experto usado en este proyecto, como se nombró en la estrategia, es un estudiante recién titulado en enfermería por la UCLM en el campus de Ciudad Real, que ha tenido la suerte de realizar prácticas en centros médicos. Sin embargo, el conocimiento que ha podido aportar es limitado si lo comparamos con el conocimiento de un pediatra que lleve trabajando, por ejemplo, 15 años, viendo día a día casos semejantes a los estudiados en el proyecto. A pesar de esto, destacar que el experto ha aportado todo aquel conocimiento que ha podido, y para el desarrollo de un proyecto de esta magnitud, ha sido más que suficiente.

Otro gran aspecto que mejoraría sin lugar a dudas el proyecto, es tener un mejor acceso a datos públicos sobre enfermedades infecciosas. Obviamente no se pueden publicar datos personales de pacientes, pero si que ayudaría a proyectos semejantes contar con datos anonimizados, de tal forma que no se identifique a las personas, ya que las diferentes técnicas de inteligencia artificial pueden ayudar a mejorar la atención a pacientes, aligerando plazos para diversas tareas como diagnóstico y tratamiento de enfermedades.

Una gran mejora para el proyecto sería el refinamiento de la etapa de preprocesado de

datos en los casos clínicos, aportando diferentes valores de gravedad para cada síntoma, como por ejemplo grave, leve, no presenta, etc. Con los datos con los que se cuenta solo se puede afirmar la presencia del síntoma en el paciente, sin especificar la gravedad del mismo. También mejoraría el proyecto ampliar el número de enfermedades infecciosas que es capaz de diagnosticar.

En cuanto a la parte de interfaz, existen dos mejoras claras. La primera, aportarle persistencia, para poder almacenar en una base de datos los usuarios de la aplicación, pacientes, diagnósticos, etc. La segunda mejora consiste en mostrar el diagnóstico en la misma interfaz, sin crear un documento de texto simple donde plasmarlo. Para ello, es necesario que Clipspy aporte la posibilidad de conocer como gestiona la salida estándar, ya que se ha considerado el redirigir la misma para mostrarla en la interfaz, sin éxito, ya que se consiguió redirigir todo menos las instrucciones lanzadas por Clipspy.

### 7.3 Competencias

Por último, se comprueban cuales de las competencias de la intensificación de computación se han adquirido gracias al desarrollo de este TFG.

- *Capacidad para tener un conocimiento profundo de los principios fundamentales y modelos de la computación y saberlos aplicar para interpretar, seleccionar, valorar, modelar, y crear nuevos conceptos, teorías, usos y desarrollos tecnológicos relacionados con la informática.*

En este proyecto se ha llevado a cabo un proceso de adquisición del conocimiento que ha ayudado a la extracción de información de un experto, para posteriormente, modelar y formalizar dicho conocimiento en un conjunto de reglas en el entorno de desarrollo CLIPS.

- *Capacidad para conocer los fundamentos, paradigmas y técnicas propias de los sistemas inteligentes y analizar, diseñar y construir sistemas, servicios y aplicaciones informáticas que utilicen dichas técnicas en cualquier ámbito de aplicación.*

Gracias al uso de la ingeniería del conocimiento y la analítica de datos, se ha conseguido desarrollar un sistema inteligente de ayuda a la decisión en enfermedades infecciosas pediátricas.

- *Capacidad para adquirir, obtener, formalizar y representar el conocimiento humano en una forma computable para la resolución de problemas mediante un sistema informático en cualquier ámbito de aplicación, particularmente los relacionados con aspectos de computación, percepción y actuación en ambientes entornos inteligentes.*

En la etapa de adquisición del conocimiento, se obtuvo el conocimiento necesario de un experto, posteriormente, siguiendo la metodología IDEAL, se formalizó y representó dicho conocimiento en reglas de producción capaces de diagnosticar enfermedades

infecciosas y generar un tratamiento para paliar los efectos de dichas enfermedades.

- *Capacidad para desarrollar y evaluar sistemas interactivos y de presentación de información compleja y su aplicación a la resolución de problemas de diseño de interacción persona computadora.*

Se ha desarrollado una interfaz capaz de ofrecer el conocimiento del sistema inteligente únicamente con su uso. De esta forma, se consigue generar una transmisión de conocimiento entre el computador y el usuario, haciendo uso de un sistema interactivo como es la interfaz de la aplicación.

- *Capacidad para conocer y desarrollar técnicas de aprendizaje computacional y diseñar e implementar aplicaciones y sistemas que las utilicen, incluyendo las dedicadas a extracción automática de información y conocimiento a partir de grandes volúmenes de datos.*

En el proceso de análisis de datos se utilizó tanto aprendizaje supervisado como no supervisado, concretando, se hizo uso de agrupación o clustering para conocer la estructura de los datos ofrecidos por la OMS, y se utilizaron árboles de decisión clasificadores para diagnosticar enfermedades infecciosas pediátricas en base a un conjunto de datos de casos clínicos.

# Referencias

- [Bec03] K. Beck. *Test-Driven Development By Example*. Addison-Wesley, 2003.
- [BMB03] R. E. Black, S. S. Morris, y J. Bryce. *Child Survival I*. The Lancet, 2003.
- [Caf21] M. Cafasso. *Clipsy Documentation*, 2021. url: <https://docplayer.net/206817484-Clipsy-documentation.html>.
- [Car87] J. D. Carrillo. *Metodología Para el Desarrollo de Sistemas Expertos*. Facultad de Informática de la Universidad Politécnica de Madrid, 1987.
- [cli15] *CLIPS Reference Manual - Volume I: The Basic Programming Guide*. CLIPS, 2015.
- [Con12] D. Conway. *Machine Learning for Hackers*. O'Reilly, 2012.
- [ECA<sup>+</sup>03] F. Escolano, M. A. Cazorla, M. I. Alfonso, O. Colomina, y M. A. Lozano. *Int. Artificial: Modelos, Técnicas y Áreas de Aplicación*. Thomson, 2003.
- [FPS96] U. Fayyad, G. Piatetsky, y P. Smyth. *The KDD Process for Extracting Useful Knowledge from Volumes of Data*. Communications of the ACM, 1996.
- [Ger17] A. Geron. *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. O'Reilly, 2017.
- [GHO] OMS. *Global Health Observatory data repository*. url: <https://apps.who.int/gho/data/node.main>.
- [GRB] R. García, B. Rossi, y P. Britos. *Metodologías de Educación de Conocimiento para la construcción de Sistemas Informáticos Expertos*. CAPIS.
- [GT16] N. M. Garzón y L. C. Torres. *Ingeniería del Conocimiento*. CICI, 2016.
- [Hig10] J. Highsmith. *Agile Project Management*. Pearson Education, 2010.
- [Hon11] K. Honavalli. *Sprint with Scrum and Get Work Done*. Carnegie Mellon University. Software Engineering Institute, 2011.

- [HV07] E. Herrera y L. E. Valencia. *Del manifiesto ágil sus valores y principios*. Universidad de las Ciencias Informáticas de La Habana, 2007.
- [Jos08] J. Joskowicz. *Reglas y Prácticas en eXtreme Programming*. Universidad de Vigo, 2008.
- [LFG13] M. Leyva, C. A. Febles, y C. J. Gulín. *Causal knowledge representation techniques: a case study in Medical Informatics*. Universidad de las Ciencias Informáticas de La Habana, 2013.
- [Mit15] R. Mitchell. *Web scraping with Python: Collecting more data from the modern web*. O'Reilly Media, 2015.
- [MP88] J. L. Mate y J. Pazos. *Ingeniería del Conocimiento: Diseño y Construcción de Sistemas Expertos*. 1988.
- [MT16] N. Miloslavskaya y A. Tolstoy. *Big Data, Fast Data and Data Lake Concepts*. Procedia Computer Science, 2016.
- [Oli00] J. A. Olivas. *Tesis Doctoral: Contribución al estudio experimental de la predicción basada en categorías deformables borrosas*. Universidad de Castilla - La Mancha, 2000.
- [Ped17] PEDIASCAPE.ORG. *PedAM: Pediatric Disease Annotations Medicines*, 2017. url: <http://www.unimd.org/pedam/>.
- [Qt21] River Bank Computing Ltd. *What is PyQt?*, 2021. url: <https://riverbankcomputing.com/software/pyqt/intro>.
- [Qtd21] The Qt Company Ltd. *Qt Designer Manual*, 2021. url: <https://doc.qt.io/qt-5/qtdesigner-manual.html>.
- [SB10] K. Schwaber y M. Beedle. *Agile Software Development with Scrum*. Pearson Education, 2010.
- [SNG16] S. Sastoque, C. Narváez, y G. Garnica. *Metodología para la construcción de Interfaces Gráficas Centradas en el Usuario*. TISE, 2016.
- [THC<sup>+</sup>16] S. R. Timarán, I. Hernández, S. J. Caicedo, A. Hidalgo, y J. C. Alvarado. *Descubrimiento de patrones de desempeño académico con árboles de decisión en las competencias genéricas de la formación profesional*. Bogotá: Ediciones Universidad Cooperativa de Colombia, 2016.

# ANEXOS



## Anexo A

# Entrevistas con el experto

## A.1 Primera entrevista

- *Fecha y hora:* 15 de marzo de 2020 a las 17:30
- *Lugar:* debido al estado de alarma, la reunión se realiza por videoconferencia.
- *Asistentes:*
  - Sergio Sevilla Ballesteros (Ing. Conocimiento)
  - Aitor Sánchez García (Experto)
- *Fuentes de conocimiento:* el experto, conocedor del mundo de la enfermería y que posee conocimientos y técnicas de actuación ante enfermedades infecciosas en niños.
- *Objetivos:* conocer que aspectos tiene en cuenta el experto en la identificación y tratamiento de una enfermedad infecciosa, para identificar el dominio y ver las posibles relaciones existentes entre estos aspectos.
- *Modo:* al ser la primera entrevista, entrevista no estructurada y abierta, para poder conocer los aspectos más importantes del dominio y ver cuál es el procedimiento que lleva a cabo el experto para tomar diferentes decisiones que lo lleven a determinadas conclusiones.
- *Planteamiento de la sesión:* se llevó a cabo una charla sin guión, de la cual, lo más destacable se puede resumir en las siguientes preguntas que le surgieron al experto durante el encuentro:
  - *Pregunta 1.* ¿Podría definir el concepto de enfermedad infecciosa?
  - *Pregunta 2.* ¿En qué se diferencia una enfermedad infecciosa de otros tipos?
  - *Pregunta 3.* ¿Cómo diferencias y clasificas las diferentes enfermedades existentes de este tipo?
  - *Pregunta 4.* ¿Cuál es la diferencia a la hora de identificar una enfermedad infecciosa en un niño de 0 a 12 años respecto a un adulto?
  - *Pregunta 5.* ¿Cuáles son las etapas y la duración de una enfermedad infecciosa?
  - *Pregunta 6.* Una vez ya sabes la enfermedad del paciente, ¿qué haces?
- *Resultados de la sesión:*

- *Respuesta 1.* Son aquellas enfermedades que son causadas por microorganismos, como pueden ser bacterias o virus, es decir, se necesitan que alguno de estos organismos entre en el organismo de un ser humano para dar lugar a la enfermedad infecciosa que estos provoquen. Por poner un ejemplo actual, la pandemia que está sacudiendo el planeta es debida al virus SARS – CoV2, y la enfermedad que genera es el COVID – 19.

Otro dato a tener en cuenta es que, si los microorganismos son capaces de transmitirse entre seres humanos, diremos que la enfermedad es contagiosa, que son las más habituales, aunque existen también enfermedades que se transmiten de otros animales al ser humano, pero entre humanos, no se puede transmitir, se les llama zoonosis.

- *Respuesta 2.* Creo que con la explicación que he dado antes se podría responder esta pregunta también, ya que la principal diferencia es que son generadas por microorganismos, pero por aportar más información, destacar que cuando haces referencia a otros tipos, te refieres a sus “antónimas”, que son las enfermedades no transmisibles, que son, por ejemplo, enfermedades crónicas. Estas enfermedades causan bastantes más muertes que las enfermedades infecciosas.
- *Respuesta 3.* Lo más importante son los síntomas, ya que tal y como el paciente te va diciendo que es lo que nota, vas descartando enfermedades hasta llegar a una enfermedad, ya que hay casos muy claros como el sarampión o la varicela, o en su defecto, si solo se puede acotar hasta un reducido grupo de enfermedades, es necesario hacer pruebas que nos confirmen la enfermedad con seguridad.
- *Respuesta 4.* La principal diferencia es que un niño quizás no sepa expresar de la misma forma y con tanto detalle cómo podría hacerlo un adulto sus síntomas, casi que depende más de lo que dicen sus padres que de lo que dice él mismo, pero la edad no influye a la hora de que una enfermedad genere unos síntomas u otros.
- *Respuesta 5.* Son 3 etapas, el periodo de incubación, que va desde que entra el microorganismo en el humano hasta que aparecen los síntomas, el período de desarrollo de la enfermedad, que abarca el periodo de tiempo donde el paciente tiene síntomas y por último el periodo de convalecencia, donde se supera la enfermedad, pero este último periodo es engañoso, porque el paciente se recupera y vuelve a hacer su rutina habitual, pero hay enfermedades infecciosas que se pueden contagiar en este periodo, lo que pone en riesgo a la gente de su entorno.
- *Respuesta 6.* Empezar un tratamiento, donde se puede mandar tomar algún fármaco que ayude a superar la enfermedad y también algunos hábitos de vida que ayuden a acabar la enfermedad.

■ *Plan de análisis.*

- Identificar los términos más relevantes.
  - Asociar algunas características a los términos.
  - Comprender la secuencia de acciones que llevar a cabo.
- *Resultados del análisis:* se han identificado los siguientes términos:
- Paciente
  - Enfermedad
  - Tratamiento.

A cada término se le han asociado las siguientes características:

- Paciente: Nombre, edad.
- Enfermedad: Síntomas.
- Tratamiento: Medicamentos y otras recomendaciones.

La secuencia de acciones que lleva a cabo el experto es:

- Tomar datos del paciente.
  - Preguntar a este por los síntomas.
  - Relacionar estos síntomas para encontrar la enfermedad del paciente.
  - Proporcionar al paciente un tratamiento para superar la enfermedad.
- *Tiempo de análisis:* el tiempo de análisis de esta sesión han sido 2 horas.
- *Recursos empleados:* los recursos usados en esta sesión han sido los ordenadores de los dos presentes y la aplicación Skype, que ha dado soporte a la videoconferencia mediante la cual se ha llevado a cabo la entrevista.
- *Otros datos complementarios:* la duración de la entrevista ha sido de una hora y cuarto.
- *Comentarios:* yo, el ingeniero de conocimiento, parto de un conocimiento nulo sobre los pasos a seguir para tratar una enfermedad infecciosa, pero como ciudadano que soy, se cuál es el proceso que se sigue con el paciente cuando acude a un médico, es por ello por lo que he deducido que un término muy evidente es el paciente y recoger sus datos.

A parte, quiero destacar que esta al ser la primera entrevista, las preguntas que he realizado son bastantes generales, porque la intención era identificar los términos generales sobre los que ahondar en próximas sesiones, pero una vez he analizado bien las respuestas dadas por el experto, algunas preguntas no han aportado información útil, ahora que ya parto de unos conceptos, las preguntas serán más estructuras y más concretas. Sobre todo, se hará hincapié en las enfermedades, para conocer más parámetros a valorar de estas más allá de los síntomas.

## A.2 Segunda entrevista

- *Fecha y hora:* 24 de marzo de 2020 a las 17:30
- *Lugar:* debido al estado de alarma, la reunión se realiza por videoconferencia.
- *Asistentes:*
  - Sergio Sevilla Ballesteros (Ing. Conocimiento)
  - Aitor Sánchez García (Experto)
- *Fuentes de conocimiento:* el experto, que además proporciona un documento sobre enfermedades infecciosas que sirve de apoyo al ingeniero del conocimiento para poder estructurar las características esenciales de las mismas.
- *Conocimiento anterior a la entrevista:* en la anterior sesión se logró identificar los conceptos claves sobre los que profundizar en esta entrevista, la enfermedad, los síntomas y su tratamiento.
- *Objetivos:* adentrarnos en el concepto de enfermedad infecciosa, ya que se considera necesario conocer que características más allá de los síntomas hay que valorar.
- *Modo:* entrevista parcialmente estructurada, ya que disponemos del conocimiento de la entrevista previa, pero no suficiente como para plantear unas cuestiones estructuradas.
- *Planteamiento de la sesión:*
  - *Pregunta 1.* Además de los síntomas, ¿qué otras características definen a una enfermedad?
  - *Pregunta 2.* ¿Por qué es importante la edad del paciente?
  - *Pregunta 3.* ¿Por qué es importante la edad del paciente?
  - *Pregunta 4.* Aparte de las características que sirven para identificar la enfermedad, ¿existen otros parámetros que los padres sean convenientes que conozcan?
  - *Pregunta 5.* ¿Hay que tener en cuenta algún aspecto a la hora de asignar un tratamiento?
  - *Pregunta 6.* ¿En qué se diferencia entonces las características adicionales de la enfermedad que usted da a los padres con las que aporta en el tratamiento?
  - *Pregunta 7.* ¿Qué aspectos favorecen o fomentan la transmisión de estas enfermedades?
- *Resultados de la sesión:*
  - *Respuesta 1.* Existen aparte de los síntomas diversos aspectos, como te dije en la anterior entrevista, los síntomas son los que casi siempre ayudan a encontrar la enfermedad del paciente, pero existen otros parámetros que en menor medida también ayudan en el proceso, por ejemplo, la edad.

- *Respuesta 2.* Porque existen enfermedades que solo aparecen en esas edades, bien por naturaleza o bien porque al niño a cierta edad se le vacuna de esa enfermedad y, por tanto, no puede padecerla, salvo fallo de vacunación, que ese es un problema mayor y que creo que se escapa de los límites planteados aquí.
- *Respuesta 3.* Si, pero van perdiendo importancia porque son muy genéricos, por ejemplo, el tiempo de incubación de la enfermedad, ya que algunos padres te pueden notificar que desde hace ciertos días han notado algo en el comportamiento de sus hijos.
- *Respuesta 4.* Depende de la gravedad de la enfermedad se informa a los padres sobre diversos parámetros que deben de tener en cuenta, se les puede informar sobre cómo se transmite la enfermedad, para que eviten algún tipo de contacto con el niño en concreto, incluso si la enfermedad es más fuerte se le dice el tiempo de exclusión, que es el tiempo que debe de estar el niño sin tener contacto con otros niños, ya que suponen un peligro para el contagio, como por ejemplo en la varicela, que es muy fácil que un niño lo transmita a otro niño, también se le pueden comentar las complicaciones que puede generar la enfermedad si se agrava, aunque a veces no es recomendable, por no meter miedo, además de que es bastante improbable si se sigue el tratamiento de que aparezcan.
- *Respuesta 5.* Por supuesto, es un apartado vital, hay que tener en cuenta las posibles patologías previas que tenga el paciente, sea de mayor o menor tamaño, ya que hay fármacos que no son compatibles. Una patología puede ser desde una alergia hasta una enfermedad crónica.
- *Respuesta 6.* Pues básicamente que las dadas en el tratamiento ayudan a superar la enfermedad, por ejemplo, si el paciente por ejemplo una simple fiebre, en el tratamiento podemos incluir un fármaco que le ayude a bajar algunas décimas y por ejemplo reposo en cama, que le ayuden a descansar y pasar mejor la enfermedad, sin embargo una característica que se puede dar a los padres sobre la enfermedad es que el tiempo de incubación de la enfermedad, que da información sobre el proceso de la enfermedad, pero no ayuda a que el paciente la supere.
- *Respuesta 7.* Múltiples aspectos y de muy diferentes ámbitos, es muy complicado controlar la transmisión de estas enfermedades, nos hemos acostumbrado a vivir con ellas, como en el caso de la gripe. Pueden ser factores sanitarios como la vacunación, ya que por ejemplo en países empobrecidos muchos niños no son vacunados y se exponen más a tener complicaciones en sus primeros años de vida. La higiene y alimentación es otro aspecto básico, sobretodo la higiene para evitar infecciones bacterianas. También por ejemplo depende del país y la atención sanitaria que tengan, el número de trabajadores, etc. También la zona de residencia, si vives en un lugar con poca densidad de población, será más difícil

que se genere un brote considerablemente grande.

■ *Plan de análisis:*

- Identificar las características esenciales de las enfermedades infecciosas.
- Valorar la importancia de cada una de estas características.
- Conocer cuáles son las enfermedades infecciosas más habituales en niños.

■ *Resultados del análisis:* se han identificado las siguientes características de las enfermedades infecciosas:

- Síntomas.
- Tiempo de incubación.
- Modos de transmisión.
- Complicaciones.

Y la ordenación de la importancia que deben de tener diferentes aspectos para identificar que enfermedad tiene el paciente, es el siguiente: 1. Síntomas, 2. Edad paciente y 3. Tiempo incubación. El resto de las características no son relevantes, solo dan información.

Como último apunte, hay que destacar que en la anterior entrevista el ingeniero justificó que había que recoger diferentes datos del paciente, y en esta entrevista se ha ratificado que está en lo cierto, ya que la edad del paciente a veces puede ayudar en el proceso de encontrar la enfermedad que padece el paciente, además, se ha añadido una característica necesaria a tener en cuenta para asignar un tratamiento al paciente, las patologías previas, ya que algún fármaco o antibiótica podría no ser assignable a un paciente debido a alguna dolencia que padezca con anterioridad.

- *Tiempo de análisis:* el tiempo de análisis de esta sesión ha sido de 1 hora y media.
- *Recursos empleados:* los recursos usados en esta sesión han sido los ordenadores de los dos presentes y la aplicación Skype, que ha dado soporte a la videoconferencia mediante la cual se ha llevado a cabo la entrevista.
- *Otros datos complementarios:* la duración de la entrevista ha sido de 45 minutos.
- *Comentarios:* destacar que en la anterior entrevista se preparan las pregunta antes de llevar a cabo la entrevista, en este caso no, el ingeniero del conocimiento solo preparó la primera pregunta, la cual dio lugar a la aparición de las características de las enfermedades infecciosas sobre las que él quería profundizar, por tanto, el resto de pregunta dependieron de la respuesta que proporcionó el experto a la primera pregunta.

### A.3 Tercera entrevista

- *Fecha y hora:* 27 de marzo de 2020 a las 11:00
- *Lugar:* debido al estado de alarma, la reunión se realiza por videoconferencia.
- *Asistentes:*
  - Sergio Sevilla Ballesteros (Ing. Conocimiento)
  - Aitor Sánchez García (Experto)
- *Fuentes de conocimiento:* el experto y el documento que aportó en la anterior sesión. Además, el experto comunica que hace uso de sus apuntes de clase.
- *Conocimiento anterior a la entrevista:* en la anterior entrevista se concretó más las características de las enfermedades infecciosas, y gracias al documento que aportó el experto, el ingeniero del conocimiento ha hecho una lista de enfermedades infecciosas.
- *Objetivos:* conocer las diferentes características de las enfermedades infecciosas de la lista que ha confeccionado el ingeniero del conocimiento.
- *Modo:* entrevista estructurada, ya que el ingeniero del conocimiento va a preguntar al experto las diferentes características de diferentes enfermedades.
- *Planteamiento de la sesión:* se acuerda con el experto que de las siguientes enfermedades aporte la información que conozca sobre: síntomas, rango de edad, formas de transmisión, tiempo de incubación y exclusión, y el tratamiento a llevar a cabo: varicela, mononucleosis, escarlatina, gripe, bronquiolitis, sarampión, enfermedad de Kawasaki, tos ferina, enterobiasis, gastroenteritis y faringoamigdalitis.
- *Resultados de la sesión:*
  - *Varicela*
    - Rango edad: 0 – 14 años.
    - Tiempo incubación: 10 – 21 días.
    - Tiempo exclusión: 5 días.
    - Síntomas principales: Piel granosa y picores.
    - Síntomas menos comunes:
    - Formas de transmisión: Contacto directo y vía respiratoria.
    - Tratamiento: Aciclovir, evitar mojar las erupciones, hidratación.
  - *Sarampión*
    - Rango edad: 0 – 5 años.
    - Tiempo incubación: 10 días.
    - Tiempo exclusión: 5 días.
    - Síntomas principales: Erupciones en la piel y fiebre muy alta.
    - Síntomas menos comunes: Problemas inmunes, tos y diarrea.
    - Formas de transmisión: Secreciones nasofaríngeas.

- Tratamiento: Reposo en cama, cuidado en la piel y antitérmicos para combatir la fiebre.
- *Bronquiolitis*
  - Rango edad: cualquier edad.
  - Tiempo incubación: 7 días.
  - Tiempo exclusión: 6 días.
  - Síntomas principales: Jadeos, fiebre alta y dificultad al respirar.
  - Síntomas menos comunes: Insuficiencia respiratoria.
  - Formas de transmisión: Vía respiratoria.
  - Tratamiento: Paracetamol, ambiente húmedo e hidratación.
- *Faringoamigdalitis*
  - Rango edad: cualquier edad.
  - Tiempo incubación: 2 - 15 días.
  - Tiempo exclusión: Innecesario.
  - Síntomas principales: Tos, fiebre, amígdalas hinchadas.
  - Síntomas menos comunes: Malestar general, amígdalas rojas.
  - Formas de transmisión: Contacto directo y vía respiratoria.
  - Tratamiento: Penicilina, reposo extendido e hidratación.
- *Mononucleosis*
  - Rango edad: 0 – 25 años.
  - Tiempo incubación: 30 – 50 días.
  - Tiempo exclusión: Innecesario.
  - Síntomas principales: Fatiga, irritación de garganta, glangios inflamados.
  - Síntomas menos comunes: Dolor de cabeza.
  - Formas de transmisión: Secreciones vías respiratorias, escasa.
  - Tratamiento: Paracetamol, descanso extendido e hidratación.
- *Enfermedad de Kawasaki*
  - Rango edad: 0 - 5 años.
  - Tiempo incubación: 0 días. Síntomas inmediatos.
  - Tiempo exclusión: Innecesario.
  - Síntomas principales: Labios rojos y con presencia de sequedad, problemas cardíacos.
  - Síntomas menos comunes: fiebre y enrojecimiento de algunas zonas de la piel.
  - Formas de transmisión: Contacto muy estrecho.
  - Tratamiento: Ácido acetilsalicílico.
- *Escarlatina*
  - Rango edad: 3 – 12 años.

- Tiempo incubación: 2 - 4 días.
  - Tiempo exclusión: 5 días.
  - Síntomas principales: Piel roja erupcionada y fiebre alta.
  - Formas de transmisión: Secreciones respiratorias.
  - Tratamiento: Amoxicilina y reposo extendido.
- *Tos ferina*
  - Rango edad: 0 – 1 años.
  - Tiempo incubación: 12 – 14 días.
  - Tiempo exclusión: Innecesario.
  - Síntomas principales: Tos, vómitos y problemas pulmonares.
  - Síntomas menos comunes: Diarrea continua.
  - Formas de transmisión: Contacto directo.
  - Tratamiento: Claritromicina, alimentación adecuada y descanso extendido.
- *Gripe*
  - Rango edad: cualquier edad.
  - Tiempo incubación: 2 - 5 días.
  - Tiempo exclusión: 6 días.
  - Síntomas principales: Fiebre normalmente alta y dolor de cabeza.
  - Síntomas menos comunes: Problemas inmunes, sensación de debilidad muscular y mareos.
  - Formas de transmisión: Secreciones respiratorias.
  - Tratamiento: Paracetamol, descanso extendido e hidratación.
- *Enterobiasis*
  - Rango edad: 0 - 12 años.
  - Tiempo incubación: 1 - 2 meses.
  - Tiempo exclusión: Innecesario.
  - Síntomas principales: Picor de ano e insomnio.
  - Síntomas menos comunes: Sensación de fatiga.
  - Formas de transmisión: Fecal - oral.
  - Tratamiento: Mebendazol e hidratación.

Lista de medicamentos y patologías incompatibles:

- Paracetamol: daños hepáticos.
- Aciclovir: alergia a la lactosa.
- Ácido acetilsalicílico: problemas cardiacos.
- Amoxicilina: diabetes.
- Mebendazol: asma.
- Claritromicina: ninguna conocida.

- Antitérmicos: ninguna conocida.

■ *Plan de análisis:*

- Identificar las características de cada una de las enfermedades infecciosas de la lista.
- Identificar el tratamiento de cada una de las enfermedades infecciosas de la lista.
- Identificar la que patologías son incompatibles con los diferentes medicamentos que formen parte del tratamiento de alguna enfermedad.
- Poder empezar con la fase de conceptualización una vez adquirido el conocimiento.

■ *Resultados del análisis:* se ha conseguido identificar cada característica de cada una de las enfermedades y sus correspondientes tratamientos, además de haber identificado las patologías incompatibles con los tratamientos, por tanto, se puede desarrollar por completo la fase de conceptualización e incluso se pueden empezar a pensar en formar reglas.

- *Tiempo de análisis:* el tiempo de análisis de esta sesión ha sido media hora.
- *Recursos empleados:* los recursos usados en esta sesión han sido los ordenadores de los dos presentes y la aplicación Skype, que ha dado soporte a la videoconferencia mediante la cual se ha llevado a cabo la entrevista.
- *Otros datos complementarios:* la duración de la entrevista ha sido de 2 horas y media.
- *Comentarios:* en esta entrevista el ingeniero del conocimiento ya sabía exactamente qué información quería del experto, es por ello por lo que no se han realizado preguntas con el fin de acotar algo, si no que al experto se le ha ido nombrado una por una la enfermedad, además de que el ingeniero le pasó una lista con las características que quería conocer de cada una.

## A.4 Cuarta entrevista

- *Fecha y hora:* 11 de julio de 2021 a las 12:00
- *Lugar:* la reunión se realiza por videoconferencia.
- *Asistentes:*
  - Sergio Sevilla Ballesteros (Ing. Conocimiento)
  - Aitor Sánchez García (Experto)
- *Fuentes de conocimiento:* el experto, conocedor del conocimiento que se solicitará a lo largo de la entrevista
- *Objetivos:* conocer en qué rango de edades aparecen las nuevas enfermedades introducidas en el sistema, así como sus posibles tratamientos.
- *Modo:* entrevista estructurada.
- *Planteamiento de la sesión:* se le plantean las siguientes preguntas al experto:
  - *Pregunta 1.* Rango de enfermedades de las enfermedades recién añadidas al sistema inteligente.
  - *Pregunta 2.* Tratamiento de las mismas.
- *Resultados de la sesión:*
  - *Respuesta 1.* El experto indica que todas las enfermedades recién añadidas pueden aparecer en cualquier momento entre los 0 y los 14 años, el rango de edad que estamos valorando como enfermedades infecciosas pediátricas. Sin embargo, si que destaca que dos de estas enfermedades si tienen un rango más limitado.  
La micosis indica que no suele aparecer en los primeros años de vida, ni en los últimos años de pediátrica, dejando la enfermedad en un rango de edad desde los 3 hasta los 10 años. Por otro lado, el impétigo, el experto indica que a partir de los 8 años, es bastante improbable que aparezca.
  - *Respuesta 2.* El experto nos indica el tratamiento de cada enfermedad por separado.
    - Malaria: Cloroquina, Artesimina e hidratación.
    - Neumonía: Penicilina, reposo e hidratación.
    - Resfriado: No hacer uso de antibióticos, reposo, hidratación y seguir una dieta blanda.
    - Micosis: Miconazol.
    - Impétigo: Tisúderma y humedecer zona afectada
    - Úlcera Péptica: Esomeprazol, dieta blanda y reposo.
    - Tuberculosis: Isoniacida y Etambutol.
    - Hepatitis A: Al no existir un tratamiento certero, reposo e hidratacion.

- Dengue: Reposo y Paracetamol en caso de fiebre alta. Uso de Tylenol en lugar de Paracetamol si el paciente presentan problemas hepáticos.
- Tifoidea: Ciprofloxacino y Ceftriaxona.
- Hipertiroidismo: Propiltiouracilo y Metimazol.
- Hipotiroidismo: Levothyroid y Synthroid.

■ *Plan de análisis:*

- Identificar el rango de edad de las nuevas enfermedades.
- Identificar los tratamientos de las nuevas enfermedades.

■ *Resultados del análisis:* el experto ha aportado el conocimiento necesario.

■ *Tiempo de análisis:* el tiempo de análisis de esta sesión ha sido de una hora.

■ *Recursos empleados:* los recursos usados en esta sesión han sido los ordenadores de los dos presentes y la aplicación Skype, que ha dado soporte a la videoconferencia mediante la cual se ha llevado a cabo la entrevista.

■ *Otros datos complementarios:* la duración de la entrevista ha sido de media hora.

## Anexo B

# Clusters obtenidos

### Países pertenecientes al cluster 0

- **Asia:** Israel, Japón y Kuwait.
- **África:** Ningún país.
- **América:** Canadá, Chile, Cuba y Estados Unidos.
- **Europa:** Austria, Bélgica, República Checa, Dinamarca, Finlandia, Francia, Alemania, Grecia, Hungría, Islandia, Irlanda, Italia, Letonia, Luxemburgo, Malta, Mónaco, Países Bajos, Noruega, Portugal, San Marino, Eslovaquia, Eslovenia, España, Suecia, Suiza, Gran Bretaña e Irlanda del Norte.
- **Oceanía:** Australia y Nueva Zelanda.

### Países pertenecientes al cluster 1

- **Asia:** Afganistán, India, Laos, Pakistán y Timor Oriental.
- **África:** Angola, Benín, Burkina Faso, Camerún, República Centroafricana, Chad, Congo, Costa de Marfil, República Democrática del Congo, Yibuti, Guinea Ecuatorial, Etiopia, Gabón, Guinea, Guinea-Bisáu, Liberia, Madagascar, Mali, Mauritania, Namibia, Niger, Nigeria, Sierra Leona, Somalia, Sudán del Sur, Sudán, Togo y Yemen.
- **América:** Haití.
- **Europa:** Montenegro.
- **Oceanía:** Ningún país.

### Países pertenecientes al cluster 2

- **Asia:** Baréin, Brunei, China, Corea del Norte, Corea del Sur, Irán, Jordania, Kazajistán, Kirguistán, Líbano, Malasia, Maldivas, Mongolia, Omán, Catar, Singapur, Sri Lanka, Siria, Tayikistán, Tailandia, Turkmenistán, Emiratos Árabes Unidos, Uzbekistán y Vietnam.
- **África:** Argelia, Egipto, Esuatini, Libia, Mauricio, Arabia Saudí, Seychelles, Sudáfrica, Trinidad y Tobago y Tunéz.

- **América:** Antigua y Barbuda, Argentina, Bahamas, Barbados, Belice, Brasil, Colombia, Costa Rica, Dominica, El Salvador, Granada, Guyana, Jamaica, México, Panamá, Paraguay, San Cristóbal y Nieves y Santa Lucía.
- **Europa:** Albania, Andorra, Armenia, Azerbaiyán, Bielorrusia, Bosnia, Bulgaria, Croacia, Chipre, Estonia, Georgia, Lituania, Polonia, Moldavia, Macedonia del Norte, Rumania, Rusia, Serbia, Turquía y Ucrania.
- **Oceanía:** Islas Cook, Fiyi, Islas Marshall, Micronesia, Nauru, Niue, Palau, Tonga y Tuvalu.

### Países pertenecientes al cluster 3

- **Asia:** Bangladesh, Bután, Camboya, Indonesia, Iraq, Myanmar, Nepal y Filipinas.
- **Africa:** Botsuana, Burundi, Cabo Verde, Eritrea, Gambia, Ghana, Kenia, Kiribati, Ruanda, Santo Tomé y Príncipe, Senegal, Uganda, Tanzania, Zambia y Zimbabue.
- **América:** Bolivia, República Dominicana, Ecuador, Guatemala, Honduras, Lesoto, Malawi, Marruecos, Mozambique, Nicaragua, Perú, Surinam y Venezuela.
- **Europa:** Ningún país.
- **Oceanía:** Papúa Nueva Guinea, Samoa, Islas Salomón y Vanuatu.

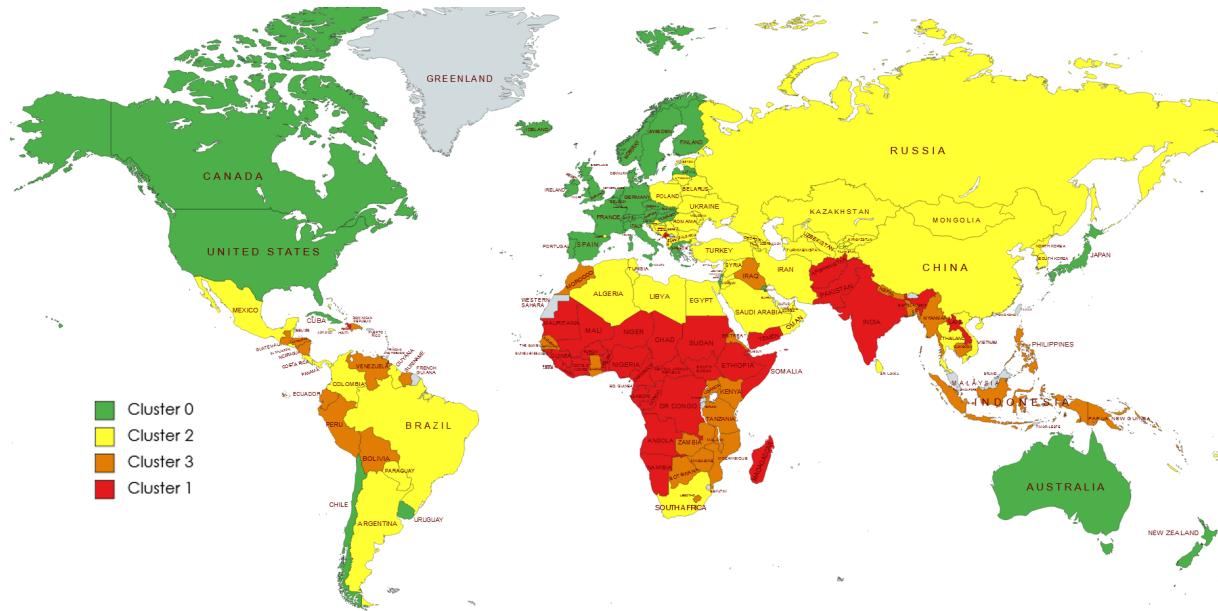


Figura B.1: Países clasificados por cluster.

## Anexo C

# Árbol de Decisión

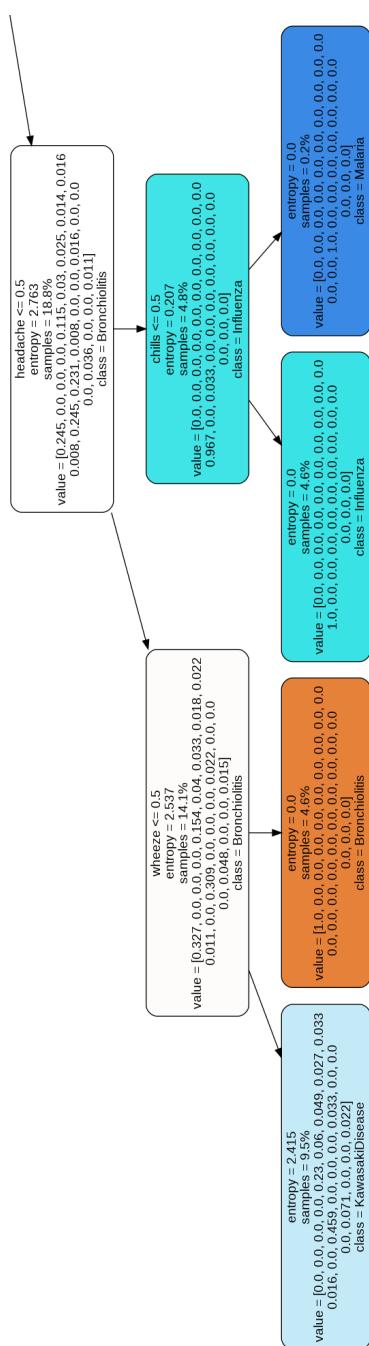


Figura C.1: Parte más izquierda del árbol.

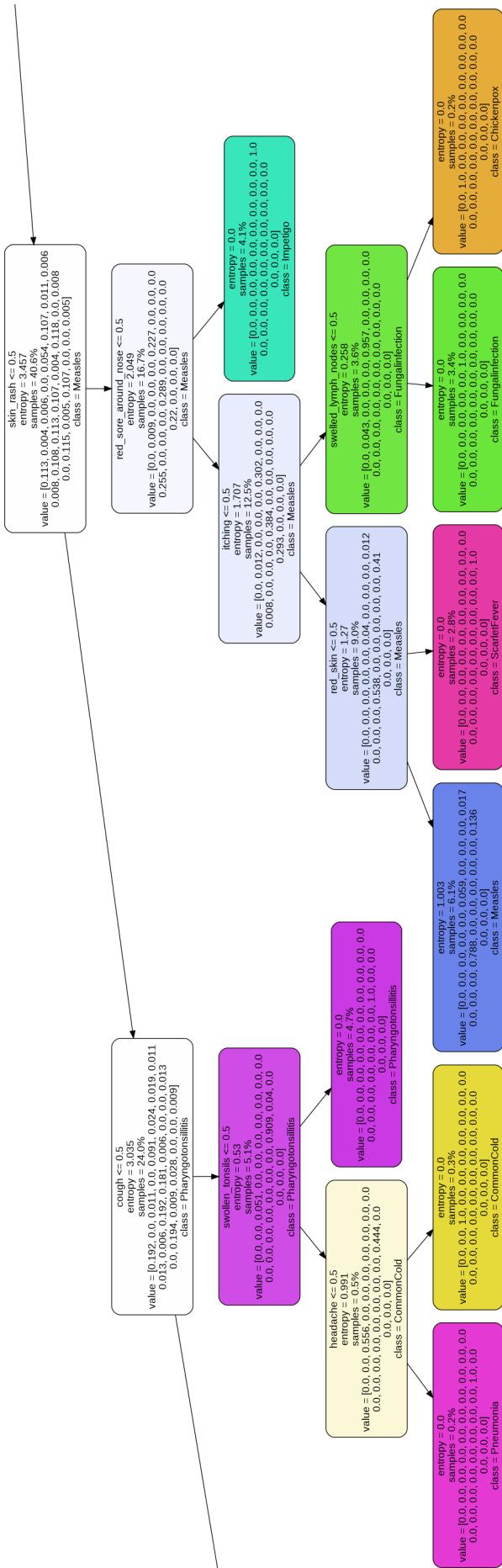


Figura C.2: Parte izquierda del árbol.

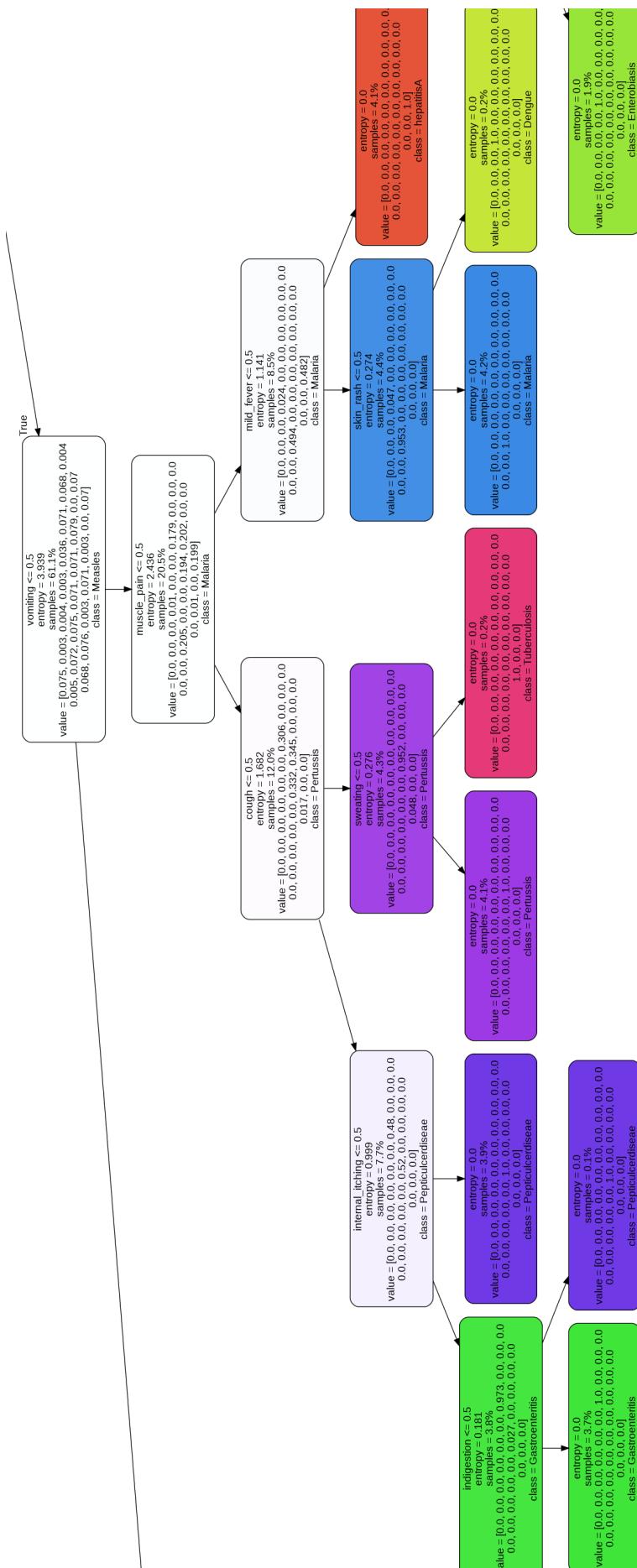


Figura C.3: Parte centro izquierda del árbol.

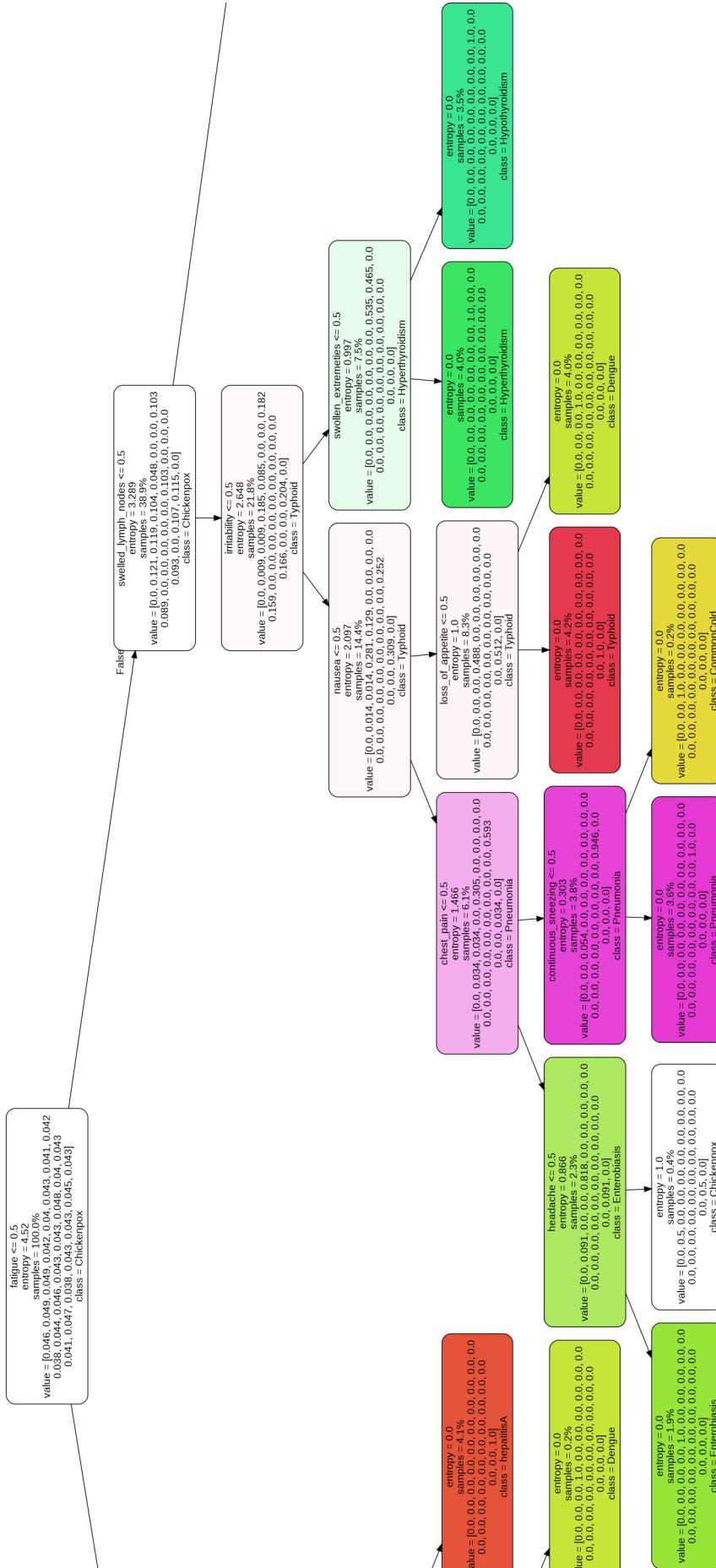


Figura C.4: Parte central del árbol.

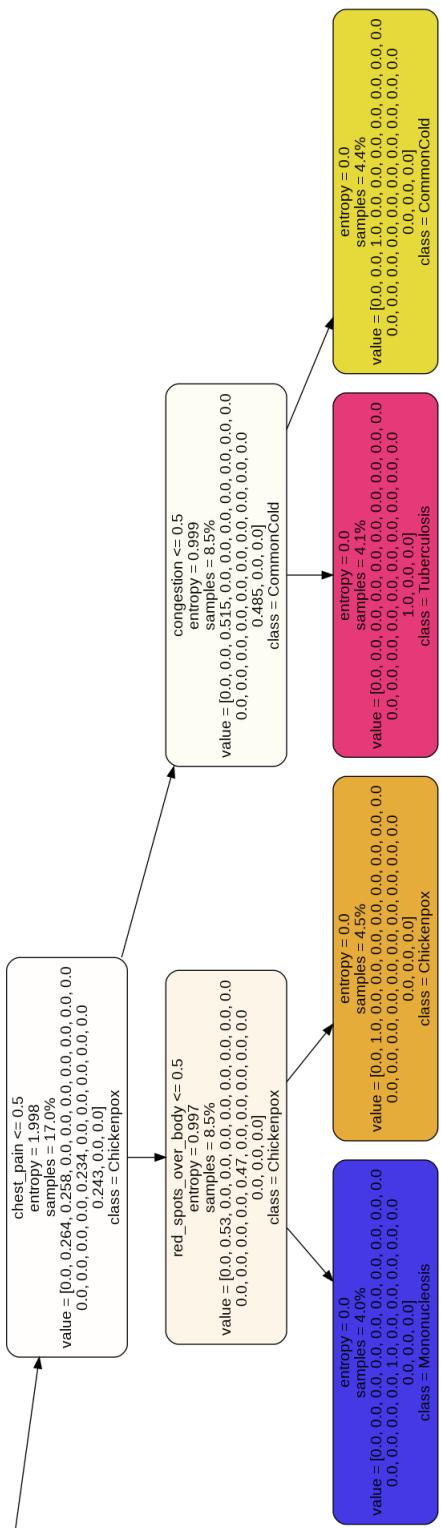


Figura C.5: Parte derecha del árbol.



## Anexo D

# Código del Sistema Experto

## D.1 Fragmento de código del sistema experto en CLIPS

```
(deftemplate paciente
  (slot nombre)
  (slot edad)
  (slot patologia))

(deftemplate sintoma
  (slot nombre)
  (slot gravedad))

;; ----- REGLAS PARA DIAGNOSTICO DE ENFERMEDADES -----
(defrule regla_enf_1
  (paciente(edad ?edad))
  (sintoma(nombre picores)(gravedad fuerte))
  (sintoma(nombre piel_granosa)(gravedad leve))
  (sintoma(nombre piel_roja)(gravedad leve))
  (test(< ?edad 14))
=>
  (assert(varicela))
  (printout t "Enfermedad diagnosticada: Varicela." crlf)
)

;; ----- REGLAS PARA RECETAR UN TRATAMIENTO -----
(defrule regla_trat_1
  (paciente(patologia ~alergia_lactosa))
  (varicela)
=>
  (assert(aciclovir))
  (assert(hidratacion))
  (printout t "Tratamiento: Aciclovir e hidratacion." crlf)
)

(defrule regla_trat_2
  (paciente(patologia alergia_lactosa))
  (varicela)
=>
  (assert(hidratacion))
  (printout t "Tratamiento: Hidratacion." crlf)
)
```

Listado D.1: Código del sistema experto en CLIPS

## D.2 Fragmento de código del sistema experto en Clipspy

```
import clips
import PyQt5

env = clips.Environment()

patologia_cb = "ninguna"
if self.cb_corazon.isChecked() == True: patologia_cb = "problemas_cardiacos"
if self.cb_hepaticos.isChecked() == True: patologia_cb = "danos_hepaticos"
if self.cb_asma.isChecked() == True: patologia_cb = "asma"
if self.cb_diabetes.isChecked() == True: patologia_cb = "diabetes"
if self.cb_alergia.isChecked() == True: patologia_cb = "alergia_lactosa"

template_string_1 = """
(deftemplate paciente
  (slot nombre (type STRING))
  (slot edad (type INTEGER))
  (slot patologia (type STRING))
)
"""

template_string_2 = """
(deftemplate sintoma
  (slot nombre_sintoma (type STRING))
  (slot gravedad (type STRING))
)
"""

env.build(template_string_1)
env.build(template_string_2)
template_1 = env.find_template('paciente')
template_2 = env.find_template('sintoma')

fact1 = template_1.assert_fact(nombre = self.le_nombre.text(), edad = int(self.cbx_edad.currentText()), patologia = patologia_cb)

ui_path = os.path.dirname(os.path.abspath(__file__)) # Conseguimos el directorio actual
env.load(ui_path + '\\BaseConocimientoClipspy.clp') ## Cargando reglas (Listado D.3)

if self.cb_g1.isChecked(): template_2.assert_fact(nombre_sintoma = 'picores', gravedad = 'fuerte')
# Resto de sintomas...
if self.cb_l30.isChecked(): template_2.assert_fact(nombre_sintoma = 'inmunosupresion', gravedad = 'leve')
env.run()
```

Listado D.2: Código del sistema experto en Clipspy

```

;; ----- REGLAS PARA DIAGNOSTICO DE ENFERMEDADES -----
(defrule regla_enf_1
    (paciente(edad ?edad))
    (sintoma(nombre_sintoma "picores")(gravedad "fuerte"))
    (sintoma(nombre_sintoma "piel_granosa")(gravedad "leve"))
    (sintoma(nombre_sintoma "piel_roja")(gravedad "leve"))
    (test(< ?edad 14))
=>
    (assert(varicela))
    (open "D:\\PedInf\\Interfaz\\diagnostico.txt" mydata "a")
    (printout mydata "Enfermedad diagnosticada: Varicela." crlf
              "La varicela es una infeccion causada por el virus zoster." crlf
              "Causa una erupcion en la piel con picazon y ampollas." crlf crlf)
    (close)
    (printout t "Enfermedad diagnosticada: Varicela." crlf)
)

(defrule regla_enf_2
    (paciente(edad ?edad))
    (sintoma(nombre_sintoma "picores")(gravedad "fuerte"))
    (sintoma(nombre_sintoma "piel_granosa")(gravedad "leve"))
    (sintoma(nombre_sintoma "ganglios_inflamados")(gravedad "leve"))
    (test(< ?edad 14))
=>
    (assert(varicela))
    (open "D:\\PedInf\\Interfaz\\diagnostico.txt" mydata "a")
    (printout mydata "Enfermedad diagnosticada: Varicela." crlf
              "La varicela es una infeccion causada por el virus zoster." crlf
              "Causa una erupcion en la piel con picazon y ampollas." crlf crlf)
    (close)
    (printout t "Enfermedad diagnosticada: Varicela." crlf)
)

;; ----- REGLAS PARA RECETAR UN TRATAMIENTO -----
(defrule regla_trat_1
    (paciente(pathologia ~"alergia_lactosa"))
    (varicela)
=>
    (assert(aciclovir))
    (assert(hidratacion))
    (open "D:\\PedInf\\Interfaz\\diagnostico.txt" mydata "a")
    (printout mydata "Tratamiento: Aciclovir e hidratacion." crlf)
    (close)
    (printout t "Tratamiento: Aciclovir e hidratacion." crlf)
)

```

Listado D.3: Reglas del sistema experto en Clipsy



Este documento fue editado y tipografiado con L<sup>A</sup>T<sub>E</sub>X empleando  
la clase **esi-tfg** (versión 0.20181017) que se puede encontrar en:  
[https://bitbucket.org/esi\\_atc/esi-tfg](https://bitbucket.org/esi_atc/esi-tfg)

[respeta esta atribución al autor]

