

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA**

CARRERA DE INGENIERÍA DE SISTEMAS

*Trabajo de titulación previo
a la obtención del título de
Ingeniero de Sistemas*

PROYECTO TÉCNICO:

**DISEÑO Y DESARROLLO DE UNA
HERRAMIENTA INTELIGENTE PARA LA
ADAPTABILIDAD DE OBJETOS DE APRENDIZAJE
CONSIDERANDO NORMATIVAS DE ACCESIBILIDAD
ENFOCADAS A PERSONAS CON DISCAPACIDAD**

AUTOR:

CHRISTIAN ROLANDO OYOLA FLORES

TUTORA:

ING. PAOLA CRISTINA INGAVÉLEZ GUERRA

CUENCA – ECUADOR

2019

CESIÓN DE DERECHOS DE AUTOR

Yo, Christian Rolando Oyola Flores con documento de identificación N°. 0302016795, manifiesto mi voluntad y cedo a la Universidad Politécnica Salesiana, la titularidad sobre los derechos patrimoniales en virtud de que soy autor del trabajo de titulación: **DISEÑO Y DESARROLLO DE UNA HERRAMIENTA INTELIGENTE PARA LA ADAPTABILIDAD DE OBJETOS DE APRENDIZAJE CONSIDERANDO NORMATIVAS DE ACCESIBILIDAD ENFOCADAS A PERSONAS CON DISCAPACIDAD**, mismo que ha sido desarrollado para optar por el título de: *Ingeniero de Sistemas*, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autor, me reservo los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, noviembre del 2019



Christian Rolando Oyola Flores

C.I.: 0302016795

CERTIFICACIÓN

Yo, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **DISEÑO Y DESARROLLO DE UNA HERRAMIENTA INTELIGENTE PARA LA ADAPTABILIDAD DE OBJETOS DE APRENDIZAJE CONSIDERANDO NORMATIVAS DE ACCESIBILIDAD ENFOCADAS A PERSONAS CON DISCAPACIDAD**, realizado por Christian Rolando Oyola Flores, obteniendo el *Proyecto Técnico*, que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana.

Cuenca, noviembre del 2019



Ing. Paola Cristina Ingavéz Guerra

C.I.: 1712214616

DECLARATORIA DE RESPONSABILIDAD

Yo, Christian Rolando Oyola Flores con documento de identificación N°. 0302016795, autor del trabajo de titulación: **DISEÑO Y DESARROLLO DE UNA HERRAMIENTA INTELIGENTE PARA LA ADAPTABILIDAD DE OBJETOS DE APRENDIZAJE CONSIDERANDO NORMATIVAS DE ACCESIBILIDAD ENFOCADAS A PERSONAS CON DISCAPACIDAD**, certifico que el total contenido de este *Proyecto Técnico* es de mi exclusiva responsabilidad y autoría.

Cuenca, noviembre del 2019



Christian Rolando Oyola Flores

C.I.: 0302016795

Agradecimientos

Quiero expresar mi agradecimiento a la Ing. Paolita Ingavélez por la confianza y la compañía brindada en el desarrollo de este proyecto, como tutora de tesis, aportándome su conocimiento y un sentimiento social hacia las personas con discapacidad, de igual manera al Dr. Vladimir Robles, quien siempre estuvo para inculcar que siempre se puede dar el mejor esfuerzo y un poco más.

Al Dr. Fernando Pesántez por abrir siempre las puertas y brindar el apoyo en proceso de investigación llevados a cabo dentro del Grupo de Investigación en Inteligencia Artificial y Tecnologías de Asistencia (GIIATA) y Catedra Unesco "Tecnologías de Apoyo a la Inclusión Educativa".

Dedicatoria

Esta dedicatoria se queda corta para los seres que estuvieron presentes en los momentos precisos y necesarios.

Seres invaluableles como mi Abuelita Hortensia, que por medio de su constante desvelo desde que tengo uso de razón proyectaron un sentimiento de responsabilidad y dedicación en mi persona.

A Cristina Flores que es madre y más que un padre, apoyándome y motivándome, junto a mi hermana Sofía en cada idea así como decisión, siendo este, uno de los resultados de sus consejos y sabiduría.

Y agradecido por los consejos y apoyo de personas fantásticas, Marco Flores V., Andres Bacuilima C., gracias por estar y por hacer el viaje más llevadero.

Abstract

The presence of ICT (Information and Communication Technologies) plays an important role in the development of the current social field and where education is not isolated from this development. Through the implementation of ICT in conjunction with main actors such as teachers and students, an educational model with a collaborative and participatory approach is developed in which pedagogical strategies are generated that allow improving access, organization and reuse of information used in processes educational, all with the aim of mediating learning and teaching in people with and without disabilities. It should be noted that the process of adaptation and accessibility of learning objects (implemented in the majority of educational spaces) characterized by their rich content, informative materials and iconic material, represents a challenge when the technical and pedagogical competencies of teachers in the face of the ICT universe to perform configuration processes on digital educational material, are not optimal enough to meet the goal of accessibility for all. In order to respond to the limitations that arise when generating an accessible learning object, it is necessary to implement strategies and systems that support this development. Therefore, accessibility guidelines and strategies that support an optimal pedagogical material that, when implemented on an educational space, become an enhancer of accessible learning environments must be taken into consideration. In this way creativity is encouraged, as well as the generation of a cooperative environment for the development of teaching and learning processes among the main actors of the training process, thus supporting traditional pedagogical practices. Based on the foregoing, this document presents a proposal that is used

as a support in the area of adaptability of educational resources for people with disabilities. This proposal focuses on facilitating teachers to develop customized components for learning variability, based on accessibility regulations in easy reading such as the UNE: 153101 and the UNE: 170002 regulations for the presentation of educational documents. The proposed proposal consists of a web environment in which learning objects are stored, analyzed and processed, which together with an expert system and application of natural language processing techniques, user needs are identified and a recommendation is given of the best configurations to be applied on the learning object to adapt it to your pedagogical needs, thus generating a new personalized educational resource. In addition, the system is characterized by the implementation and training of language models that allow incorporating new features that provide the ability to enhance the system to the educational needs of different disability profiles.

Keywords: Accessibility, Learning Objects, Easy Reading, Natural Language Processing, Regular Expressions

Índice general

Índice de figuras	XIII
Índice de cuadros	XX
1. Resumen	1
2. Introducción	3
3. Objetivos	6
3.1. General	6
3.2. Específicos	6
4. Fundamentos y trabajo relacionado	8
4.1. Objetos de aprendizaje	8
4.1.1. Características de un Objeto de Aprendizaje	10
4.2. Metodología base para el diseño, creación y evaluación de objetos de aprendizaje	11
4.2.1. Fase de análisis	11

ÍNDICE GENERAL	x
4.2.2. Fase de diseño	12
4.2.3. Fase de implementación	13
4.2.3.1. Metadatos	13
4.2.4. Fase de evaluación	14
4.2.5. Fase de publicación	14
4.3. Diseño multimedial del Objeto de Aprendizaje	15
4.3.1. Exelearning como herramienta de autor para la creación de contenidos educativos	15
4.4. Estructura interna de un Objeto de Aprendizaje	16
4.5. Ensamblaje de un Objeto de Aprendizaje	18
4.6. UNE 153101:2018 EX - Lectura Fácil: Pautas y recomendaciones para la elaboración de documentos	18
4.6.1. Proceso de trabajo para la adaptación de un documento a Lectura Fácil	20
5. Diseño y desarrollo de la propuesta	27
5.1. Especificaciones técnicas del proyecto	27
5.1.1. Especificaciones de hardware	27
5.1.2. Especificaciones de software	28
5.1.3. Herramientas de desarrollo empleadas	29
5.2. Diagrama de Caso de Uso	34
5.3. Diagramas de Secuencia	39
5.4. Diagrama objeto relacional del sistema	40

ÍNDICE GENERAL	XI
5.5. Arquitectura general de la propuesta planteada	42
5.5.1. Capas del sistema	44
5.5.1.1. Capa de extracción de información	44
5.5.1.2. Capa de sistema experto	48
5.5.1.2.1. Definición del sistema de gestión:	50
5.5.1.2.2. Base de hechos:	56
5.5.1.2.3. Base de conocimientos:	60
5.5.1.2.4. Motor de inferencias:	64
5.5.1.2.5. Interface con el Experto:	66
5.5.1.2.6. Interface con el usuario:	67
5.5.1.3. Capa de procesamiento:	69
5.5.1.3.1. Entrenamiento del modelos de lenguaje spaCy:	72
5.5.1.4. Análisis de contrastes:	79
5.5.1.5. Capa de ensamblaje	84
6. Experimentación y resultados	90
6.1. Plan de experimentación	90
Conclusiones	135
Recomendaciones	137
Trabajo futuro	138

ÍNDICE GENERAL	XII
Bibliografía	139
Apéndice A. Presentación de propuesta	141

Índice de figuras

4.1. Características de un Objeto de Aprendizaje	10
4.2. Estructura lógica interna de un componente .html desarrollado en eXeLearning	16
4.3. Países participantes en PISA-D para adultos	23
4.4. Resultados de PISA-D en estudiantes de 15 años del Ecuador	24
5.1. Herramientas principales implementadas	29
5.2. Estructura de las capas del sistema inteligente	34
5.3. Proceso de recomendación de configuraciones basados en las necesidades del alumno	39
5.4. Determinación de necesidades para definición del perfil de usuario	40
5.5. Diagrama relacional del sistema experto	41
5.6. Estructura de las capas del sistema inteligente	43
5.7. Ejemplo de estructura de O.A exportado en sitio web mediante eXeLearning	44
5.8. Ejemplo de estructura DOM de un recurso .html perteneciente a un O.A desarrollado en eXeLearning	45
5.9. Resultado de scraping web en archivos html del O.A	47

5.10. Arquitectura básica del sistema experto desarrollado [50](#)

5.11. Arquitectura general del sistema de gestión definido para el sistema experto desarrollado [50](#)

5.12. Arquitectura general del motor de inferencia empleado en el sistema experto. [65](#)

5.13. Interface con el experto para el ingreso de hechos remotos [67](#)

5.14. Interface con el usuario (Docente) donde se presenta el formulario de necesidades. [68](#)

5.15. Interface con el usuario (Docente) donde se presenta las recomendaciones dadas por el S.E. [68](#)

5.16. Interface con el usuario (Docente) donde se presenta la descripción de la recomendación. [68](#)

5.17. Ficheros resultates de capa de extracción [69](#)

5.18. Extracción recursiva de contenido de ficheros [70](#)

5.19. Identificación de ID's de configuraciones resultantes del S.E [70](#)

5.20. Estructuras de datos centrales de spaCy [72](#)

5.21. Estructura de predicción para modelos de entrenamiento spaCy [73](#)

5.22. Etiquetado de parte del discurso (POS) y reconocimiento de entidad con nombre (NER) [80](#)

5.23. Ejemplo de un texto original extraído del O.A previo procesamiento [80](#)

5.24. Presentación de los resultados del S.E aplicadas sobre los textos extraídos del O.A [80](#)

5.25. Resultado de aplicación de recomendaciones dadas por el S.E sobre texto del O.A [81](#)

5.26. Resultado de proceso de adaptación previo a ensamblaje.	81
5.27. Screen de recursos del O.A a partir de navegador sin cabecera para análisis de contrastes.	83
5.28. Resultado de identificación de contrastes en componente html con su correspondiente escala de colores.	83
5.29. Escala de colores contraste identificando color, porcentaje de color y cantidad de pixeles por color	84
5.30. Escribir y descargar EPUB posterior validación de textos analizados por Docente	88
5.31. Resultado Final de procesamiento del sistema inteligente	89
6.1. Interfaz para carga de O.A. por Docente	94
6.2. Identificación de componentes Html de O.A.	95
6.3. Resultado del S.E para experimentación N°1 para discapacidad Cognitiva .	95
6.4. Análisis de Legibilidad en textos originales extraídos del O.A.	96
6.5. Análisis de Perspicuidad en textos originales extraídos del O.A.	96
6.6. Análisis de Lecturabilidad en textos originales extraídos del O.A.	97
6.7. Análisis de Crawford en textos originales extraídos del O.A.	97
6.8. Ejemplo de presentación de Índices de Legibilidad sobre O.A	98
6.9. Análisis de legibilidad de textos extraídos del O.A. post implementación del procesamiento del sistema inteligente	99
6.10. Análisis de Perspicuidad de textos extraídos del O.A. post procesamiento del sistema inteligente	99

6.11. Análisis de Lecturabilidad de textos extraídos del O.A. post procesamiento sistema inteligente	<u>100</u>
6.12. Análisis de Crawford de textos extraídos del O.A. post procesamiento del sistema inteligente	<u>100</u>
6.13. Análisis de legibilidad lingüística de textos originales	<u>101</u>
6.14. Análisis de legibilidad lingüística de textos originales	<u>101</u>
6.15. O.A. Evaluación Primaria del Trauma Infantil Procesado	<u>102</u>
6.16. Análisis de legibilidad lingüística de textos originales	<u>102</u>
6.17. Análisis de legibilidad lingüística de textos procesados	<u>103</u>
6.18. O.A. Evaluación Primaria del Trauma Infantil Procesado	<u>103</u>
6.19. Análisis de legibilidad lingüística de textos originales	<u>104</u>
6.20. Análisis de legibilidad lingüística de textos procesados	<u>104</u>
6.21. O.A. Sistemas numéricos	<u>105</u>
6.22. Análisis de legibilidad lingüística de textos originales	<u>105</u>
6.23. Análisis de legibilidad lingüística de textos procesados	<u>106</u>
6.24. O.A. Mitosis	<u>106</u>
6.25. Análisis de legibilidad lingüística de textos originales	<u>107</u>
6.26. Análisis de legibilidad lingüística de textos procesados	<u>107</u>
6.27. O.A. Proceso de la digestión	<u>108</u>
6.28. Análisis de legibilidad lingüística de textos originales	<u>108</u>
6.29. Análisis de legibilidad lingüística de textos procesados	<u>108</u>
6.30. O.A. Estructura del núcleo de la célula eucariota	<u>109</u>

6.31. Análisis de legibilidad lingüística de textos originales	<u>109</u>
6.32. Análisis de legibilidad lingüística de textos procesados	<u>110</u>
6.33. O.A. Técnica de medición de la temperatura corporal	<u>110</u>
6.34. Análisis de legibilidad lingüística de textos originales	<u>111</u>
6.35. Análisis de legibilidad lingüística de textos procesados	<u>111</u>
6.36. O.A. Cromosomas	<u>112</u>
6.37. Análisis de legibilidad lingüística de textos originales	<u>112</u>
6.38. Análisis de legibilidad lingüística de textos procesados	<u>113</u>
6.39. O.A. Óxidos	<u>113</u>
6.40. Análisis de legibilidad lingüística de textos originales	<u>114</u>
6.41. Análisis de legibilidad lingüística de textos procesados	<u>114</u>
6.42. O.A. Estructura de un sistema de Telemedicina	<u>115</u>
6.43. Análisis de legibilidad lingüística de textos originales	<u>115</u>
6.44. Análisis de legibilidad lingüística de textos procesados	<u>116</u>
6.45. O.A. Sistema solar	<u>116</u>
6.46. Análisis de legibilidad lingüística de textos originales	<u>117</u>
6.47. Análisis de legibilidad lingüística de textos procesados	<u>117</u>
6.48. O.A. Lógica matemática	<u>118</u>
6.49. Análisis de legibilidad lingüística de textos originales	<u>118</u>
6.50. Análisis de legibilidad lingüística de textos procesados	<u>119</u>
6.51. O.A. Algoritmos	<u>119</u>

6.52. Análisis de legibilidad lingüística de textos originales	120
6.53. Análisis de legibilidad lingüística de textos procesados	120
6.54. O.A. Teoría de conjuntos	121
6.55. Análisis de legibilidad lingüística de textos originales	121
6.56. Análisis de legibilidad lingüística de textos procesados	122
6.57. O.A. Organización y Arquitectura del Computador	122
6.58. Análisis de legibilidad lingüística de textos originales	123
6.59. Análisis de legibilidad lingüística de textos procesados	123
6.60. O.A. Introducción Salud sexual y reproductiva	124
6.61. Análisis de legibilidad lingüística de textos originales	124
6.62. Análisis de legibilidad lingüística de textos procesados	125
6.63. O.A. Traumatismos	125
6.64. Análisis de legibilidad lingüística de textos originales	126
6.65. Análisis de legibilidad lingüística de textos procesados	126
6.66. O.A. Procesador	127
6.67. Análisis de legibilidad lingüística de textos originales	127
6.68. Análisis de legibilidad lingüística de textos procesados	128
6.69. Resultado de configuraciones para perfil con discapacidad auditiva	128
6.70. Análisis de legibilidad lingüística de textos procesados	129
6.71. Análisis de legibilidad lingüística de textos procesados	129
6.72. Análisis de legibilidad lingüística de textos procesados	130

6.73. Análisis de legibilidad lingüística de textos procesados	130
6.74. Análisis de legibilidad lingüística de textos procesados	131
6.75. Análisis de legibilidad lingüística de textos procesados	131
6.76. Análisis de legibilidad lingüística de textos procesados	132
6.77. Análisis de legibilidad lingüística de textos procesados	132
6.78. Análisis de legibilidad lingüística de textos procesados	133
6.79. Análisis de legibilidad lingüística de textos procesados	133
A.1. Presentación de propuesta a miembros del Consejo Nacional para la Igualdad de Discapacidades (Conadis), 2019	141
A.2. Presentación de propuesta a experto español de Discapacidad, 2019	141

Índice de cuadros

5.1. Acceso al sistema.	35
5.2. Carga de O.A.	35
5.3. Definición del perfil del estudiante a adaptar el O.A.	36
5.4. Aplicación de la configuraciones planteadas por el Sistema Experto	36
5.5. Verificación de las configuraciones aplicadas	37
5.6. Ensamblaje de N.O.A.	37
5.7. Despliegue del N.O.A. sobre un LMS	38
5.8. Definición de pautas y recomendación que conforman la base de conocimiento del Sistema Experto	59
5.9. Escala Delta E [26]	82
6.1. Grados de escala μ [1]	91
6.2. Nivel de perspicuidad (P) [18]	92
6.3. Escala de lecturabilidad (L) [6]	92
6.4. Experimentación con perfiles de discapacidad.	93

6.5. Formulario de necesidades presentada al Docente responsable de adaptar el O.A.....	<u>94</u>
--	---------------------------

Capítulo 1

Resumen

La presencia de las TIC (Tecnologías de Información y comunicación) juega un papel importante en el desarrollo del ámbito social actual y donde la educación no se ve aislada de este desarrollo. Mediante la implementación de las TIC en conjunto con actores principales como los docentes y el alumnado, se desarrolla un modelo educativo con enfoque colaborativo y participativo en el cual se generan estrategias pedagógicas que permitan mejorar el acceso, organización y reutilización de la información empleada en procesos educativos, todo ello con el objetivo de mediar el aprendizaje y enseñanza en personas con y sin discapacidad.

Cabe señalar que el proceso de adaptación y accesibilidad de objetos de aprendizaje (implementados en la mayoría de espacios educativos) característicos por su contenido rico en textos informativos y material icónico, representa un reto cuando las competencias técnicas y pedagógicas de los docentes frente al universo de las TIC para desempeñar procesos de configuración sobre material educativo digital, no son lo suficientemente óptimos para cumplir con el objetivo de accesibilidad para todos.

Con el propósito de responder a las limitantes que se presentan al momento de generar un objeto de aprendizaje accesible, es necesaria la implementación de estrategias y de

sistemas que brinden soporte en este desarrollo. Por ello, se debe tomar en consideración pautas y estrategias de accesibilidad que respalden un óptimo material pedagógico que al ser implementado sobre un espacio educativo, se convierta en un potenciador de ambientes de aprendizaje accesibles. De esta forma se fomenta la creatividad, así también como la generación de un ambiente de cooperación para el desarrollo de procesos de enseñanza y aprendizaje entre los actores principales del proceso formativo, apoyando de esta manera las practicas pedagógicas tradicionales.

Por lo expuesto anteriormente, en este documento se presenta una propuesta que se emplea como soporte en el área de la adaptabilidad de recursos educativos para personas con discapacidad, dicha propuesta se enfoca en facilitar a los docentes el desarrollo de componentes personalizados para una variabilidad de aprendizaje, tomando como base normativas de accesibilidad en lectura fácil como la UNE:153101 y la normativa UNE:170002 para la presentación de documentos educativos.

La propuesta planteada consta de un entorno web en el cual se almacenan, analizan y procesan objetos de aprendizaje, que en conjunto con un sistema experto y aplicación de técnicas de procesamiento de lenguaje natural, se identifican las necesidades de los usuarios y se brinda una recomendación de las mejores configuraciones a ser aplicadas sobre el objeto de aprendizaje para adaptarlo a sus necesidades pedagógicas, generando así un nuevo recurso educativo personalizado.

Además, el sistema se caracteriza por la implementación y entrenamiento de modelos de lenguaje que permiten incorporar nuevas características que brinden la capacidad de potenciar el sistema a las necesidades educativas de diferentes perfiles de discapacidad.

Palabras clave: Accesibilidad, Objetos de Aprendizaje, Lectura Fácil, Procesamiento de Lenguaje Natural, Expresiones regulares

Capítulo 2

Introducción

La UNESCO en el 2016 hace referencia al problema de estudio y señala que “aproximadamente 40 millones de los 115 millones de niños y niñas que tienen discapacidad en el mundo no han accedido a instituciones educativas” [\[24\]](#). En el Ecuador es posible evidenciar según cifras del CONADIS y el INEC, que del 100 % de la población ecuatoriana con algún tipo de discapacidad entre 4 y 17 años, únicamente el 12.14 % tiene la oportunidad de acceder a una institución y proceso educativo [\[12\]](#). Por lo mencionado, en el Ecuador el derecho a una educación inclusiva ha sido establecido en el marco legal ecuatoriano, a pesar de lo cual existe una escasa aplicación de los medios necesarios en las instituciones educativas, para cumplir con esta normativa.

La educación inclusiva se refiere a la inserción de todos los niños, niñas y adolescentes, con y sin discapacidad, aprendiendo juntos en las distintas instituciones educativas regulares con medios apropiados según sus diversas necesidades.

Actualmente con el objetivo de apoyar el desarrollo de una educación y formación inclusiva, se ha masificado la adopción de Objetos de Aprendizaje (O.A) en procesos educativos por parte de docentes dentro de las diferentes instituciones educativas (primaria, secundaria

y superior). Siendo éstos recursos educativos digitales, los cuales gracias a su estructura constituida por meta-datos y diferente granularidad, permiten fomentar el desarrollo de competencias específicas por parte de los estudiantes regulares y con discapacidad, posibilitando desarrollar habilidades y ser participes en sus procesos educativos, garantizando su reutilización en diversos contextos donde responden a un patrón de necesidad, considerando características pedagógicas y tecnológicas.

En combinación con estos recursos educativos, surgen pautas que permiten apoyar la generación de material educativo inclusivo, entre las cuales sobresale la normativa UNE 153101:2018 EX de Lectura Fácil, por ser un elemento facilitador de comprensión lectora a personas que presentan dificultades en esta habilidad y entre las cuales se encuentran colectivos de personas con discapacidad [23]. Dicha normativa responde a una preocupación de UNESCO (1963) por ofrecer textos sencillos de comprender a personas con baja alfabetización y de esta forma facilitar el aprendizaje de un oficio [23].

La lectura fácil es una de las soluciones que progresivamente ha comenzado a implantarse para hacer que los contenidos educativos sean accesibles a todos. En algunos países como en Suecia, la lectura fácil cuenta con un importante grado de desarrollo. En España han comenzado a promover experiencias en este sentido que han sido muy importantes, tanto por el esfuerzo de adaptación como por el impacto obtenido.

“La lectura Fácil cubre una necesidad social que permite garantizar el derecho de acceso a la información escrita en igualdad de condiciones” [23].

“Dentro de dicha normativa se recoge en detalle la descripción metodológica para la adaptación, creación y validación de documentos en Lectura Fácil, así como las pautas y recomendaciones para la redacción y diseño de los mismos” [23].

Por lo expuesto en lo anterior, este trabajo busca contribuir en el proceso de creación y adaptación de Objetos de aprendizaje accesibles y reutilizables, mejorando la integración

de dichos recursos educativos en procesos de instrucción inclusiva y accesibles orientados a la educación virtual, eliminando barreras que impiden garantizar la comprensión del contenido escrito, contribuyendo en el diseño para todos y la igualdad de oportunidades en la participación.

Capítulo 3

Objetivos

3.1. General

Diseñar y desarrollar una herramienta inteligente para la adaptabilidad de objetos de aprendizaje considerando normativas de accesibilidad enfocadas a personas con discapacidad.

3.2. Específicos

- OE1. Estudiar la Normativa experimental española UNE 153101:2018 EX para la adaptabilidad de texto e ilustraciones en el contexto de la Lectura Fácil y contrastes cro-máticos, orientado a colectivos de personas con discapacidad.
- OE2. Diseñar y desarrollar un módulo basado en Procesamiento del Lenguaje Natural (PLN) orientado a la generación de resúmenes y frases simples a partir de textos contenidos en el Objeto de Aprendizaje, adaptando complementos para-textuales.
- OE3. Diseñar y desarrollar un módulo de análisis de contraste automatizado capaz de identificar las mejores prácticas de accesibilidad para el uso de fondos e ilustraciones

en el Objeto de Aprendizaje, adaptado para personas con discapacidad, empleando para ello técnicas de visión por computador.

- OE4. Diseñar y desarrollar un módulo para generar reportes de adaptabilidad de meta-datos resultante del proceso de re-ensamblaje del Objeto de Aprendizaje.
- OE5. Diseñar y ejecutar un plan de experimentación para la validación de la propuesta planteada con una muestra de al menos 30 objetos de aprendizaje (muestra estadística por conveniencia).

Capítulo 4

Adaptabilidad de Objetos de Aprendizaje: Fundamentos y trabajo relacionado

4.1. Objetos de aprendizaje

Como se ha venido evocando en el desarrollo de este trabajo, es un hecho que el uso de las TICs y la variedad de características que apoyan al desarrollo educativo han permitido adaptar una era digital en la transmisión de conocimiento y con ello llegar a modificar una metodología de enseñanza tradicional y su forma de presentarse a los usuarios finales.

Existe una amplia variedad de definiciones que tratan de inferir un concepto de qué es un Objeto de Aprendizaje (O.A.) y es erróneo cerrar la idea a únicamente un solo concepto.

Basados en los acontecimientos de 1994 en el cual Wayne Hodgins, emplea por primera vez el término “Objeto de Aprendizaje”, inspirado al observar a su hijo jugar con bloques de LEGO, nos atrevemos a adoptar, para parecer de los autores, una definición que abarca

ampliamente varios detalles y características de lo que constituye un O.A basado en la idea de Wayne, dicho concepto hace referencia a una “Unidad Didáctica digital independiente, cuya estructura está formada por un objetivo de aprendizaje específico, un contenido, un conjunto de actividades y una autoevaluación ” [\[17\]](#).

Al definir un O.A como una “Unidad Didáctica” se puede interpretar como un conjunto “integrado, organizado y secuencial de elementos con sentido propio” [\[17\]](#), por el cual se proyecta el resultado de una investigación o un estudio a estudiantes objetivo.

Al hacer referencia a “Digital”, se adopta una idea y tratamiento frente a un recurso de software educativo o software de enseñanza a ser empleado durante un proceso determinado de aprendizaje y por un tiempo definido.

Al decir que es “Independiente” se comprende de que puede ser utilizado durante un proceso educativo sin emplear un tercer material ya que gracias a su estructura, este O.A “puede cumplir con un objetivo educativo simple” [\[17\]](#)

Gracias a los meta-datos se puede conocer el tipo de contenido y propósito educativo por el que fue desarrollado el O.A sin tener la necesidad de acceder en su totalidad al contenido del mismo, estos meta-datos son un factor importante ya que permiten además simplificar el uso y gestión de su contenido en distintos entornos educativos así como agilizar su búsqueda y localización en algún repositorio. Por medio de un nivel de reutilización de un O.A se puede determinar si es óptimo o no para el desarrollo de unidades de conocimiento útiles en la generación de nuevos módulos, o para el desarrollo de nuevo material educativo a una mayor escala además, esta característica de los O.A es la responsable de señalar si un material educativo determinado puede ser implementado en un entorno diferente en el que fue desarrollado y ser desplegado sin inconvenientes.

4.1.1. Características de un Objeto de Aprendizaje

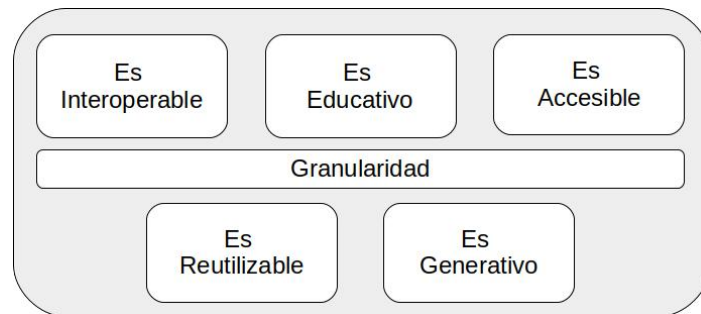


Figura 4.1 Características de un Objeto de Aprendizaje.

Interoperable: es la característica que le permite a un O.A. ser desplegado en diferentes entornos educativos y tecnológicos con un estándar homogéneo de accesibilidad y asegurar su total funcionalidad dentro de dicho entorno [\[15\]](#).

Educativo: es la cualidad principal y objetiva con la que cuenta un O.A., ya que dicha característica debe facilitar la comprensión de un concepto o teoría, así como promover el desarrollo de habilidades y competencias dentro de un grupo con enfoque social cultural, tecnológico, cognitivo, entre otros [\[15\]](#).

Accesible: facilidad para ser identificados, localizados a través de un etiquetado con diverso descriptores (meta-datos) que permitirán la catalogación y almacenamiento en un determinado repositorio [\[15\]](#).

Reutilizable: es una de las característica más relevantes de un O.A. ya que es la que permite determinar si dicho recurso puede o no ser empleado en distintos contextos, ya sean estos tecnológicos o educativos, a más de determinar si es apto para la adaptación de nuevos componentes educativos que permitan enriquecer el contenido del O.A. original [\[15\]](#).

Generativo: “capacidad para construir nuevas lecciones, unidades, módulos, entre otros, a partir de su ensamblaje con otros Objetos de Aprendizaje. Posibilidad de actualización

o modificación por medio de la colaboración de la comunidad de desarrollo de Objetos de Aprendizaje, aumentando sus potencialidades” [\[15\]](#).

Granular: se lo puede traducir como un margen de adaptación a las necesidades o a los perfiles de aplicación sobre el cual el O.A se va a desarrollar, asegurando además una mayor facilidad para la reutilización de este, al cumplir con esta característica se habla de granularidad fina [\[15\]](#).

4.2. Metodología base para el diseño, creación y evaluación de objetos de aprendizaje

La metodología que se define a continuación fue la base para definir la estructura sobre la cual la mayoría de los objetos de aprendizaje se desarrollan, siendo esta un punto de partida para definir la estructura homogénea que comparten los O.A., dicha metodología consta de 5 fases que determinan las mejores practicas al momento de desarrollar un O.A. Cabe indicar que esta metodología fue el resultado de “19 metodologías de factor utilizadas en Iberoamérica, mismas que han sido ampliamente utilizadas en el diseño y creación de material educativo digital” [\[15\]](#), misma que sirve de guía para aquellos instructores que son los responsable de llevar todo un proceso de desarrollo y que no cuentan con el apoyo de un equipo especializado en el área de la creación de material educativo digital.

4.2.1. Fase de análisis

Durante esta fase se identifica la necesidad y los usuarios finales, para el cual es desarrollado el O.A. Para esta etapa es clave un levantamiento de información para determinar perfiles de usuario así como las estrategias a ser empleadas durante el desarrollo de los contenidos. Para esta fase se puede emplear una matriz de necesidades la cual sugiere [\[15\]](#):

- Identificar el tema del O.A.
- Describir textualmente el contenido.
- Identificar el nivel educativo de la población objetivo (Primaria, Secundaria, Universidad, Otros).
- Establecer el perfil del estudiante en términos de estilos de aprendizaje.
- Tiempo para abordar el O.A completo
- Identificar los elementos y factores que favorecen el proceso de enseñanza dentro y fuera de un entorno digital.
- Establecer el tipo de licencia a utilizar.
- Identificar requerimientos técnicos como plugins necesarios, navegadores, entre otros.

4.2.2. Fase de diseño

Para el desarrollo de esta fase es importante trabajar en el diseño multimedial y en aspectos referentes al diseño instruccional, es decir, se genera un esfuerzo tanto desde la perspectiva educacional como desde la perspectiva tecnológica. Al igual que en la fase de análisis se define una plantilla para el diseño de un O.A [\[15\]](#).

- Diseño instruccional
 - Describir textualmente el contenido.
 - Definir el objetivo de aprendizaje.
 - Definir los contenidos.
 - Definir las actividades.

- Generar un proceso de
- autoevaluación. Diseño multimedial
 - Diseño de la interfaz.
 - Estructura de las plantillas.
 - Navegación.

4.2.3. Fase de implementación

En este proceso interviene la incorporación de etiquetas conocidas como meta-datos, que permiten localizar al O.A. dentro de un repositorio determinado o un ROA (Repositorio de Objetos de Aprendizaje), gracias a la definición de estas etiquetas se enriquece la propiedad de accesibilidad al conocer el objetivo y el perfil de usuario para el cual fue desarrollado el O.A.

4.2.3.1. Metadatos

Los metadatos los podemos interpretar como un conjunto de datos que permiten realizar una descripción “técnica” de otros datos, esto con el objetivo de reconocer o identificar un Objeto de Aprendizaje.

Gracias a esta característica con la que cuenta un O.A. es posible la unificación de un criterio válido para la búsqueda de un O.A. dentro de un Repositorio de Objetos de Aprendizaje (ROA).

Actualmente existen varios estándares para metadatos siendo los principales:

- Dublin Core Metadata Initiative (DCMI) que cuenta con 15 etiquetas para describir el material digital.

- Learning Object Metadata (LOM) con 9 etiquetas principales de las cuales se ramifican muchas más.
- Metadata Learning Resource (MLR) siendo este el más empleado para la definición de anotaciones semánticas y el más popular en los estándares de metadatos.

4.2.4. Fase de evaluación

Desde el punto de vista del estudiante o del usuario final, es importante determinar si el O.A. generado cumple con el objetivo de ser fácil de usar, útil pedagógicamente y si es perceptiblemente positivo. Y desde el punto de vista del docente la fase de evaluación le permite identificar si el material que se desarrollo cumple con las expectativas académicas y pedagógicas, a más de ello “valorar su efectividad tecnológica y didáctica potencial del O.A.” antes de ser empleado en la vida real [\[15\]](#).

4.2.5. Fase de publicación

Esta fase se la relaciona directamente con el tipo de licenciamiento que aplica el O.A., teniendo en consideración los contenidos como fotografías, vídeos, animaciones, entre otros recursos, incorporando además las condiciones de uso que se aplican para la distribución del material desarrollando dentro de la comunidad interesada. Entre las licencias más comunes podemos encontrar :

- Copyright: se debe brindar un permiso por parte de los creadores para el uso y distribución del material [\[15\]](#).
- Copyleft: al realizarse algún cambio o adaptación con el objetivo de generar un nuevo versionado del material, este resultado final debe ser distribuido con las mismas licencias y bajo los mismos términos o condiciones de uso [\[15\]](#).

- Creative Commons (CC): dentro de esta licencia se ramifican diferentes configuraciones, pero casi todas comparten el mismo principio de distribución de material otorgando cierta libertad para la reproducción, adaptación de nuevo contenido y hasta la manera de citar, bajo ciertas restricciones [\[15\]](#).

4.3. Diseño multimedial del Objeto de Aprendizaje

La estructura del diseño multimedial está compuesta por diversas ramas del diseño en las cuales interviene diseño gráfico, diseño editorial, diseño web y además diseño multimedia. Cada rama aporta su ciencia para enriquecer al O.A. de forma que sea llamativa y brinde una mejor experiencia educativa a los estudiantes, a más de ello con el diseño multimedial se logra que el entorno donde se desarrolla la comunicación entre el O.A. y el alumno sea más usable y accesible.

4.3.1. Exelearning como herramienta de autor para la creación de contenidos educativos

“eXeLearning es una herramienta de código abierto (open source) que facilita a los docentes la creación y publicación de contenidos educativos en soportes informáticos pudiendo inclusive llevarlo a generar sitios web completos” [\[15\]](#).

Mediante esta herramienta los autores pueden generar una estructura base y sólida que se comporta como el esqueleto del O.A., gracias a esta herramienta se pueden generar las pantallas las cuales albergan los recursos textuales, gráficos, de audio y vídeo, además test interactivos de evaluación que son accesibles desde un entorno web. Mediante la característica de embeber (insertar un contenido determinado elaborado en un lenguaje de programación dentro de otro lenguaje de programación) se pueden incorporar contenidos nativos ya sea de

HTML, JavaScript, estilos CSS, fórmulas, y muchos más. En eXeLearning estos elementos se conocen como “i-devices”.

Gracias a esta herramienta se puede definir el diseño de interfaz con hojas y estilos en cascada (CSS) que en conjunto con páginas web (HTML) conforman el esquema técnico de un O.A. (Ver Figura 4.2), siendo este el esquema sobre el cual se desarrolla el core de nuestra propuesta.

```

40     <li><a href="elaboracin.html" class="no-ch">Elaboración</a></li>
41     <li><a href="proceso_de_simpificacin.html" class="no-ch">Proceso de Simplicación</a></li>
42   </ul>
43 </li>
44 </ul>
45 </li>
46 <li><a href="actividades.html" class="no-ch">Actividades</a></li>
47 <li><a href="autoevaluacin.html" class="no-ch">Autoevaluación</a></li>
48 <li><a href="bibliografa.html" class="no-ch">Bibliografía</a></li>
49 </ul>
50 </div>
51 <div id="topPagination">
52 <div class="pagination noprt">
53 <a href="index.html" class="prev"><span><span>&laquo; </span></span>Previous</span></a> <span class="sep"
54 </div>
55 </div>
56 <div id="main-wrapper">
57 <div id="main"><a name="main"></a>
58 <div id="nodeDecoration"><h1 id="nodeTitle">Introducción</h1></div>
59 <div class="iDevice_wrapper preknowledgeIdevice em_iDevice" id="id13">
60 <div class="iDevice_emphasis1" >
61 <div class="iDevice_header"><h2 class="iDeviceTitle">Álgebra Booleana</h2></div>
62 <div class="iDevice_inner">
63 <div class="iDevice_content_wrapper">
64 <div id="ta13_55_2" class="block iDevice_content">
65 <p><span style="font-size: medium;">El Álgebra de Boole permitió establecer la analogía entre los
66 ello:</span></p>
67 <ul>
68 <li><span style="font-size: medium;">El álgebra Booleana o de Boole, consiste en un método para re
69 strong> y a tres operadores: <strong>AND (y), OR (ó) y NOT (no).</strong></span></li>
70 <li><span style="font-size: medium;">El álgebra Booleana se emplea en la construcción de computado
71 <li><span style="font-size: medium;">Es fundamental destacar que los sistemas modernos trabajan a
72 <li><span style="font-size: medium;">En términos generales, el álgebra de Boole se puede emplear a
73 <li><span style="font-size: medium;">Las computadoras modernas realizan sus operaciones y almacena
74 <li><span style="font-size: medium;">Se emplea en el diseño de circuitos electrónicos.</span></li>
75 <li><span style="font-size: medium;">Se emplea ampliamente en la programación (condiciones lógicas
76 </ul>
77 <span style="font-size: medium;">Te invitamos a ver este video elaborado por RTÉ Ireland's Nat

```

Figura 4.2 Estructura lógica interna de un componente .html desarrollado en eXeLearning.

4.4. Estructura interna de un Objeto de Aprendizaje

La estructura básica de un O.A. se la puede dividir en 4 bloques importantes que el responsable de la generación de dicho material debe definir en primera instancia para que el resultado final sea el adecuado (metodología Dicrevoa 2.0), siendo estos:

- Introducción: se realiza una presentación del O.A, esto con el objetivo de familiarizar y dar un breve acercamiento sobre el tema a tratar a los estudiantes [\[16\]](#).
- Objetivos: mediante esta apartado se definen el alcance que va a tener el material presentado, a más de ello se define las circunstancias sobre las que se va a desempeñar el estudiante [\[16\]](#), ej.:
 - Conocer el álgebra Booleana y sus operadores para la resolución de problemas de lógica.
- Contenidos: para la definición de los contenidos se deben tomar en cuenta 3 aspectos importantes:
 - La selección de los contenidos: se define el material relevante a incluir dentro del O.A., esto con el objetivo de dar información puntual sobre la temática a desarrollarse, ligada específicamente a los objetivos planteados [\[16\]](#).
 - Organización de los contenidos: esto se refiere a la estructura en la que se va a presentar los contenidos a los estudiantes, esta puede ser de forma jerárquica o por tareas y sub tareas [\[16\]](#).
 - Presentación de los contenidos: este aspecto final es el responsable generar una atención en detalle del O.A. por parte de los estudiantes, y por ende se debe generar una presentación de la información de manera coherente y específica, integrando además recursos seleccionados para resaltar el material final [\[16\]](#).
- Actividades: mediante la incorporación de información didáctica con aspectos de retroalimentación permite generar en el usuario un aprendizaje más profundo, que luego es puesto en práctica para la solución de situaciones o problemas reales, ligados a la temática tratada [\[16\]](#).

- Autoevaluación: mediante este aspecto se brinda un espacio para la reflexión sobre el contenido aprendido y de refuerzo sobre el material que no fue aprendido, permitiendo al usuario reforzar dicho contenido en base a una retroalimentación y brindando la opción de un nuevo intento para validar lo aprendido [\[16\]](#).

4.5. Ensamblaje de un Objeto de Aprendizaje

Al momento de hablar de ensamblaje podemos referirnos al nivel de granularidad de un O.A., ya que esta propiedad es la responsable de determinar si el material desarrollado puede ser reutilizado o ensamblado en otro O.A. de similar contexto o si este nivel de granularidad es el adecuado para el desarrollo de una colección de O.A. que juntos conforman módulos, evaluaciones, entre otros.

4.6. UNE 153101:2018 EX - Lectura Fácil: Pautas y recomendaciones para la elaboración de documentos

“La lectura fácil se dirige a un grupo de destinatarios muy amplio, generalmente personas que por diferentes causas presentan dificultades en la comprensión lectora, y entre ellas se encuentran colectivos de personas con discapacidad. Así la lectura fácil, como elemento facilitador de esta comprensión, permite garantizar el ejercicio de derechos recogidos en la Convención sobre los derechos de las Personas con Discapacidad que fue ratificada por España en 2008” [\[23\]](#).

Definiendo como eje principal facilitar y garantizar la comprensión lectora y escrita, la normativa UNE:153101 EX surge como pautas y recomendaciones que especifican una

metodología para la adaptación, creación y validación de documentos en lectura fácil, dividido en 3 bloques principales [\[23\]](#).

- Redacción del texto: en este bloque se detallan diferentes pautas en relación a signos ortográficos, siglas, uso de conjugaciones, y muchos más, entre las cuales tenemos:
 - Pautas y recomendaciones relacionadas con la ortotipografía.
 - Pautas y recomendaciones relacionadas con el vocabulario y las expresiones
 - Pautas y recomendaciones relacionadas con frases y oraciones.
 - Pautas y recomendaciones relacionadas con la organización del texto y el estilo.
- Diseño/maquetación de documentos: en este bloque se describen diferentes recomendaciones a la hora de presentar un documento escrito, teniendo en cuenta contrastes de color entre las letras y los fondos de los documentos, estilos de textos, entre otros. Dentro de este bloque se definen diferentes pautas tales como:
 - Pautas y recomendaciones relacionadas con la presentación del documento.
 - Pautas y recomendaciones relacionadas con las imágenes.
 - Pautas y recomendaciones relacionadas con los complementos paratextuales.
 - Pautas y recomendaciones relacionadas con los complementos paratextuales icónicos.
- Validación de la comprensibilidad de los documentos: en este bloque final se definen distintas recomendaciones a adoptar por un grupo seleccionado de validadores con el objetivo de determinar en sesiones cuan comprensible es el texto adaptado a la Lectura Fácil, entre estas recomendación se definen:
 - Organización de la lectura.
 - Introducción del tema a los validadores.

- Comprensión del vocabulario.
- Comprensión del texto.

El producto final de la implementación de las pautas planteadas, se traduce en un documento en Lectura Fácil, este material puede ser implementado en diferentes entornos ya sean para apoyar los planes educativos o netamente informativos, para un grupo variado de lectores, pero el desarrollo de esta normativa está pensado con énfasis en personas con algún tipo de dificultad de comprensión lectora, entre las cuales se puede mencionar [\[23\]](#):

- Discapacidad intelectual.
- Trastorno mental.
- Trastornos específicos del lenguaje.
- Daño cerebral sobrevenido.
- Dificultades de aprendizaje de la lectura.
- Trastorno del espectro autista.
- Trastorno por déficit de atención con o sin discapacidad.
- Discapacidad auditiva.
- Sordo-ceguera.
- El proceso de envejecimiento.
- Bajo nivel de alfabetización.
- Desconocimiento del idioma.
- Dificultades cognitivas y lingüísticas
- Otras circunstancias.

4.6.1. Proceso de trabajo para la adaptación de un documento a Lectura Fácil

Para el desarrollo de un documento de lectura fácil intervienen varios actores, entre los cuales podemos encontrar [\[23\]](#):

- El adaptador: persona que adecúa un documento a Lectura Fácil.

- Diseñador - maquetador: persona que aporta creatividad al documento basado en las pautas y recomendaciones de redacción y diseño del documento.
- Dinamizador - validadores: persona que gestiona la fase de validación.

Y se compone de 2 fases principales que son [\[23\]](#):

- Fase de adaptación (adaptador, diseñador/maquetador): dentro de esta fase se realizan las siguientes tareas:
 - Recoger las indicaciones del promotor y consensuar el proyecto de adaptación.
 - Recopilar los datos sobre el documento original que se va a adaptar.
 - Analizar el contenido y la estructura del texto a adaptar.
 - Organizar la información y tomar decisiones sobre los datos analizados en base a los objetivos del documento.
 - Elaborar el borrador del documento adaptado.
 - Comprobar si el contenido del borrador en Lectura Fácil se ajusta al propósito del documento original.
 - Facilitar al dinamizador el borrador en Lectura Fácil y el documento original.
- Fase de validación (validadores, dinamizador/es): para la fase de validación se realizan las siguientes actividades:
 - El dinamizador recibe el documento en L.F y las indicaciones del promotor.
 - El dinamizador planifica y dirige las sesiones de validación.
 - Los validadores intervienen en el proceso.
 - El dinamizador recoge los aportes de los validadores.

- El dinamizador realiza un informe para el adaptador/diseñador/maquetador con las propuestas de modificación.
- Se realizan las adaptaciones correspondientes.
- Se confirman la incorporación de los cambios presentados.
- Se genera un informe final de la validación.

Mediante la adaptación de esta normativa para el desarrollo de documentos se genera un aporte significativo para asegurar que personas con y sin discapacidad pueden acceder información escrita superando las barreras de comprensión e interacción por este tipo de medios a la información.

Herramientas existentes para la adaptabilidad de Objetos de Aprendizaje: una breve descripción general En el año 2018 PISA (Programa para la Evaluación Internacional de Estudiantes) publicó los resultados acerca de la participación de Ecuador y otros 8 países como Bután, Camboya, Guatemala, Honduras, Panamá, Paraguay, Senegal y Zambia (Ver Figura 4.3), en la medición de las habilidades en tres áreas de aprendizaje: ciencias, lectura y matemáticas (Ver Figura 4.4), orientada a estudiantes de 15 años, con el objetivo de determinar si este grupo de estudiantes cuenta con las competencias básicas para tener una plena participación dentro de la sociedad y examinar hasta que punto pueden extrapolar los conocimientos adquiridos en entornos dentro y fuera de las instituciones educativas [\[11\]](#).

PISA-D (PISA para el desarrollo) fue la evaluación específica que se desarrolló en Ecuador, siendo esta una evaluación adaptada y orientada a países de economías pequeñas y medianas, reflejando a más de ello, diferentes situaciones de pobreza así también como desventajas socioeconómicas [\[11\]](#).



Figura 4.3 Países participantes en PISA-D [11]

- “Entre los países que participaron en PISA-D, Ecuador tuvo el mejor desempeño en todas las materias evaluadas obteniendo un resultado dentro del promedio” [11].
- Y en el área de lectura, en Ecuador “ el 50 % de los estudiantes encuestados tuvieron un nivel de desempeño inferior al nivel 2 en lectura” [11].
- “En lo que respecta la variación de resultados dentro del país, el desempeño medio en lectura es inferior de los hombres que al de las mujeres, la diferencia es de 8 puntos. ” [11].
- “En Ecuador, el 25 % de la población estudiantil con el nivel socioeconómico más bajo tiene una probabilidad 3 veces mayor de tener un nivel de desempeño menor al nivel 2 en todas las áreas evaluadas en PISA-D [11].

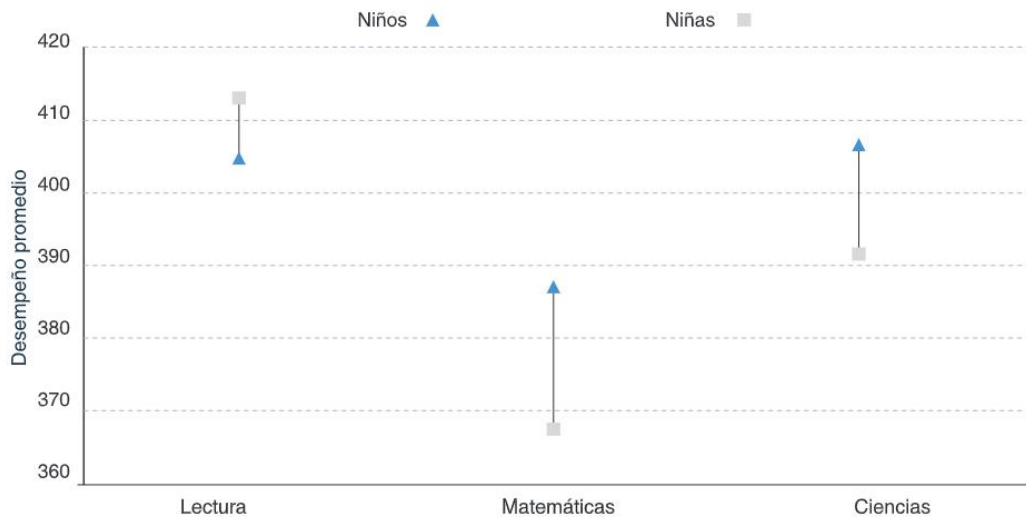


Figura 4.4 Resultados de PISA-D en estudiantes de 15 años del Ecuador [11]

Estos resultados reflejan que a pesar de que nuestro país se encuentra dentro del promedio de la región, “el desempeño regional se encuentra en niveles bajos” [11].

Este índice refleja una existencia baja de competencias básicas en los estudiantes del Ecuador al respecto de comprensión lectora, y por ende surge la preocupación sobre la proyección de dichos resultados en niños y niñas con discapacidad, demostrando quizá una falta de compendio de la administración pública con respecto a promover la generación de estrategias para desarrollar un panorama decente en niñas y niños con y sin discapacidad.

En base a los antecedentes planteados e identificando los conceptos de un O.A, sus características, potencial y facilidad de adaptación en entornos de aprendizaje, así también como las pautas que los docentes pueden adoptar para la generación de material educativo rico en usabilidad y generar un gran impacto en el desarrollo pedagógico, podemos dar una breve revisión a herramientas inteligentes y de soporte que se han generado para apoyar a las estrategias educativas basadas en el potencial de un O.A. y normativas de accesibilidad enfocadas en niños, niñas y jóvenes con y sin discapacidad.

Es así como un grupo de investigadores desarrollan un sistema inteligente para generar automáticamente resúmenes de vídeo para objetos de aprendizaje accesibles enfocado en personas con pérdida auditiva, que tiene como objetivo la traducción de subtítulos accesibles en Lengua de Señas Ecuatoriano, proponiendo una metodología de presentación ágil de contenido educativo multimedia, brindando con esta herramienta una estrategia para la traducción de los subtítulos de vídeos (Closed Caption), a una Lengua de Señas Ecuatoriano, para un usuario con pérdida de audición discapacitante [\[8\]](#).

En una línea de análisis de accesibilidad, con el empleo de modelos de conocimiento y mediante el planteamiento y adaptación de modelos de ontologías tales como WCAG 2.0, Schema, DRD y SKOS, se ha desarrollado una herramienta que permite la evaluación de objetos de aprendizaje, enfocada en los metadatos de accesibilidad relacionadas a las discapacidades tales como intelectual, física, visual y auditiva, mediante esta propuesta se han generado varios planes de evaluación sobre O.A. empleados en procesos educativos, demostrando que el sistema infiere de manera óptima en base a determinados criterios de éxito, cuales de los diferentes metadatos empleados en los O.A. afectan y favorecen a las discapacidades estudiadas, beneficiando de esta manera a la usabilidad y la adaptabilidad de material educativo a personas con y sin discapacidad [\[9\]](#) [\[10\]](#).

Con el objetivo de brindar apoyo al diagnóstico o prevención de diferentes trastornos, mediante el uso de las TICs, se han desarrollado diferentes aplicativos y estrategias enfocadas en determinadas áreas, es así que con esta misma misión surge una propuesta de un ecosis-tema basado en herramientas TIC inteligentes para apoyar el diagnóstico y la intervención de pacientes con trastornos de la comunicación. Esta propuesta interactúa con 3 factores relevantes para su funcionalidad, que es un aplicativo móvil el que se encarga de interactuar con aquellos pacientes que sufren una discapacidad motora mediante ejercicios fonatorios, actividades motoras orofaciales, así también como la construcción de oraciones y juegos que intervienen con procesos de terapia de comunicación. Un asistente robótico que interactúa

mediante el reconocimiento facial con los pacientes por medio de sonidos, actividades de relajación, entre otras actividades, y una sala multisensorial para apoyar a la estimulación temprana, auditiva, táctil y visual. Incorporando además técnicas de I.A (Inteligencia artificial), para el desarrollo de planes de terapia enfocados a expertos en el tratamiento de trastornos de comunicación [\[14\]](#).

Capítulo 5

Diseño y desarrollo del sistema inteligente para la adaptabilidad de Objetos de Aprendizaje en base a la normativa UNE 153101: EX

5.1. Especificaciones técnicas del proyecto

Con el objetivo de generar una replica futura del sistema se brinda una guía de los requerimientos técnicos para el mismo.

5.1.1. Especificaciones de hardware

Se pone a consideración que el sistema se desplegó en dos entornos, de desarrollo y de producción:

- Servidor de pruebas: un computador portátil Lenovo G40-70 hizo el rol de servidor y base de desarrollo del sistema.
 - Procesador: Intel(R) Core(TM) i7-4510U CPU @ 2.00GHz
 - Almacenamiento: 931GiB (1TB)
 - Ram:12GiB
- Servidor de producción: una instancia de rendimiento ampliable AWS-T2 small fue la base de despliegue del sistema en producción.
 - Procesador: 1vCPU
 - Almacenamiento: 25GB
 - Ram:2GB

5.1.2. Especificaciones de software

- Servidor de pruebas:
 - Sistema Operativo: la fase de desarrollo fue llevada a cabo en un entorno Linux 4.15.0-29deepin-generic, con Deepin 15.4 x64 como S.O, distribución China basada en Debian, característico por un solido sistema de actualizaciones y compatibilidad con aplicaciones, así como su entorno de escritorio amigable, cruciales para brindar fluidez en el desarrollo y despliegue del sistema.
- Sistema Producción:
 - Sistema Operativo: en la fase de producción se trabajó sobre Ubuntu 18.04 X64 de igual manera que en desarrollo es basado en Debian, asegurando un soporte y procesos de actualización de su repositorio, a más de ello su compatibilidad

con diferentes bibliotecas del sistema, generando un entorno sólido de despliegue (LTS).

5.1.3. Herramientas de desarrollo empleadas

El core o núcleo del sistema se constituye de dos lenguajes sólidos y mediante un canal de comunicación entre estos dos lenguajes, se logra cumplir con los objetivos planteados con esta propuesta. En este entorno “multilenguaje” intervienen varias herramientas (Ver Figura 5.1):



Figura 5.1 Herramientas principales implementadas (Fuente: Autor)

- Java Enterprise Edition (EE): consiste en un conjunto de estándares que permiten el desarrollo de aplicaciones web empresariales basados en una arquitectura distribuida, dichas aplicaciones son escritas en Java y son desplegadas en un servidor de

aplicaciones. En conjunto con estos estándares, se emplean diferentes bibliotecas como:

- Java Server Faces: JSFv2 proporciona un conjunto de componentes que se definen al estilo de etiquetas en las páginas XHTML con el apoyo de Facelets, en si JSF es un framework que se basa en Servlets para proporcionar dichas etiquetas, característico por su modelo vista controlador. La principal funcionalidad de JSF es proporcionar plantillas para facilitar la creación de componentes compuestos y a más de generar un soporte para el procesamiento de las peticiones HTTP, permitiendo convertir y validar los datos de entrada en combinación con las llamadas a métodos del controlador [\[5\]](#).
- OmniFaces: esta es una librería desarrollada para solucionar algunos inconvenientes que se puedan presentar en JSF2 proporcionando algunos componentes y no precisamente visuales como [\[13\]](#):
 - Remarcado de campos para procesos de validación.
 - Soporte para componentes HTML5 específicos.
 - Exception Handler.
 - Conversión de objetos select o drop-downs.
 - Validadores multicampo.
 - Y también ofrece funciones al momento de manipular arreglos, fechas, conversiones, entre otros.
- Jsoup: desarrollado por Jonathan Hedley el cual es gerente del equipo de AWS AI (desde su perfil 2019), Jsoup es una biblioteca de código libre de Java, funciona como analizador de HTML desde una URL, un archivo seleccionado o una carpeta (estructura para análisis de nuestros O.A.) similar a la funcionalidad de JQuery y selectores CSS, permite navegar por la estructura DOM (Modelo de

Objeto de Documento) para la búsqueda y extracción de datos contenidos en los documentos HTML o XML [7].

- Python / Django: Python en combinación con Django conforman el core del sistema. Python (desarrollado por Guido van Rossum, inspirado por Flying Circus de Monty Python) por una parte es un lenguaje de programación interpretado orientado a objetos, aparte de su particular sangría rigurosa, Python permite a los desarrolladores la legibilidad de su código en varios sistemas operativos (Unix, Mac OS, OS/2, entre otros). Por otra parte Django es un framework escrito en Python, el cual contiene un conjunto de componentes listos para usar, que hacen del planeta un lugar menos estresante, en el se encuentran un conjunto de recursos que permiten desplegar (en base a estándares de programación de python) de manera más ágil y directa los proceso de registro, control de sesiones, panel administrativo, formularios, carga de archivos, y varios recursos que se les puede sacar provecho. En síntesis Django permite el desarrollo de contenido para la interpretación por parte de un servidor web (Apache) [20].

De la misma manera en combinación con Python y Django se hace uso de distintos componentes para la funcionalidad de los módulos descritos más adelante:

- Selenium: mediante selenium se expone una API sencilla para desplegar pruebas con el uso de selenium webdriver (“from selenium import webdriver”), que básicamente permite enviar y ejecutar determinados comandos de python a diferentes navegadores, permitiendo desplegar un sitio web para verificar si su estructura es correcta, activar la funcionalidad de un botón, entre otras características.

Descargar: <https://n9.cl/m57>

- Selenium phantomjs: para explotar el potencial de selenium implementamos un navegador headless (sin cabeza), que básicamente es un navegador sin interfaz gráfica de usuario. Con PhantomJs como headless crawling se puede generar

un entorno de recuperación de data mining, scraping web, y de información determinada.

Descargar: <https://n9.cl/5839>

- DEPOT: DEPOT permite generar un canal de almacenamiento y ser empleado fácilmente en entornos Web, mediante Depot se genera un almacenamiento en determinados backends como Disco local.

Descargar: <https://n9.cl/dg5>

- textacy: en una biblioteca de procesamiento de lenguaje natural que es empleada en conjunto con spaCy, basado con los fundamentos de tokenización, análisis de dependencias, entre otros. Mediante textacy podemos realizar un análisis y normalización y texto sin procesar, identificando palabras, ngramas, entidades, términos clave y otros elementos determinados.

Descargar: <https://n9.cl/0gq>

- RegEx Module: “Este módulo proporciona operaciones de coincidencia de expresiones regulares similares a las que se encuentran en Perl”. [19] Con el uso de RegEx se puede realizar la verificación de cadenas dentro de un patrón para búsqueda determinada, para trabajar con python se hace uso de un paquete incorporado re.

Descargar: <https://n9.cl/nfys>

- Clips: “El Sistema de Producción Integrada de Lenguaje C (CLIPS) es un lenguaje de programación basado en reglas útil para crear sistemas expertos y otros programas donde una solución heurística es más fácil de implementar y mantener que una solución algorítmica. Escrito en C para portabilidad, CLIPS se puede instalar y usar en una amplia variedad de plataformas.” [2]

Descargar: <https://n9.cl/s130>

- spaCy: es una biblioteca de software libre para el procesamiento de lenguaje natural escrito en Python y Cython, mediante los modelos estadísticos de redes neuronales ofrece soporte multilingüe para inglés, español, francés, portugués, entre otros. Se diferencia de algunas bibliotecas de Procesamiento de lenguajes natural (PLN), en su uso para sistemas de producción, para la identificación de entidades, dependencias, etiquetado de discursos, empleando para ello redes neuronales convolucionales. Para uso educativo permite implementar un árbol de análisis de dependencias, que se puede generar a partir de un visualizador DisplaCy [21].

Descargar: <https://n9.cl/c19s>

- OpenCV: es una potente librería open source empleada en procesos de visión por computador para determinadas áreas como por ejemplo el análisis y procesamiento de imágenes, identificación y reconocimiento de objetos y de rostros, detección de movimientos, clasificación de objetos, entre muchas otras funciones más con propósitos específicos. OpenCV (Open Source Computer Vision) cuenta con un gran repositorio de algoritmos que permiten desarrollar dichos procesos, ahorrando tiempo y esfuerzos de desarrollo [22].

Descargar: <https://n9.cl/c19s>

5.2. Diagrama de Caso de Uso

Docentes, Instructores: encargados de adaptar el O.A. a un recurso accesible de Lectura Fácil mediante el sistema inteligente propuesto los docentes o instructores tienen la capacidad de cargar, procesar, editar y generar un nuevo O.A. adaptado a la normativa UNE:153101 (Ver Figura 5.2).

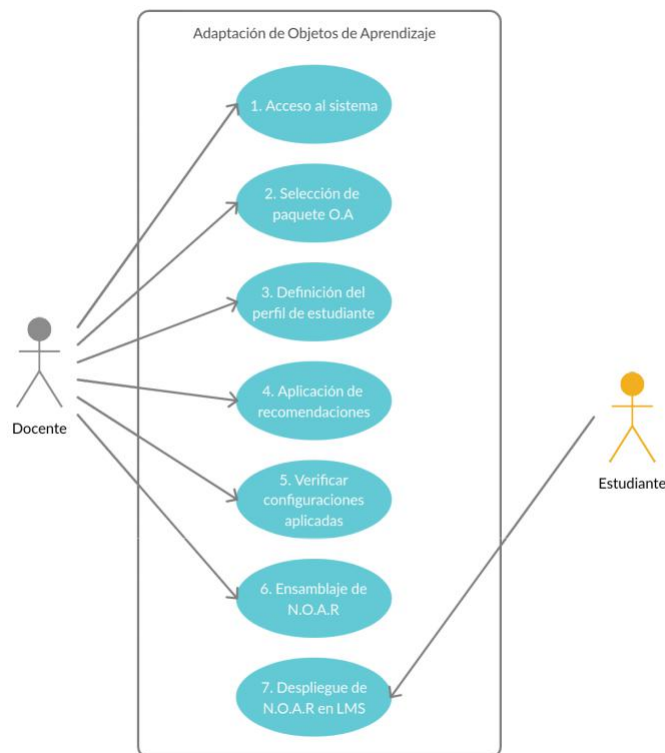


Figura 5.2 Estructura de las capas del sistema inteligente (Fuente: Autor)

El docente accede al entorno web y en el apartado de carga del O.A. para adaptar el contenido educativo procede a seleccionar el path del O.A., al identificar este recurso web se presenta un formulario para identificar el perfil del usuario al cual se va a adaptar el O.A., posterior a esta acción el docente procede a aplicar las recomendaciones propuestas por el Sistema Experto. A continuación a esta implementación, el docente es el encargado de verificar las adaptaciones aplicadas y tiene la libertad de realizar un proceso extra de edición

del Nuevo Objeto de Aprendizaje (N.O.A.) y proceder a desplegarlo sobre un LMS. En los cuadros presentados a continuación se puede apreciar en detalle los actores así como el flujo de acciones y condicionales que intervienen en la funcionalidad del sistema:

Nombre	1. Acceso al Sistema
Actor	Docente
Descripción	Acceso al entorno Web Administrativo
Precondiciones	Ninguna
Flujo Normal	1. Ingreso a entorno web del sistema
Flujo Alternativo	
Post Condición	

Cuadro 5.1 Acceso al sistema.

Nombre	2. Selección de paquetes O.A.
Actor	Docente
Descripción	Selecciona paquete de O.A. a cargar al sistema
Precondiciones	1. Actor ingresado al sistema
Flujo Normal	1. Ingreso a entorno web del sistema 2. Seleccionar paquete de O.A.
Flujo Alternativo	
Post Condición	Presentación de formulario para determinar perfil de usuario

Cuadro 5.2 Carga de O.A.

Nombre	3. Definición del perfil del estudiante
Actor	Docente
Descripción	El Docente determina las necesidades del usuario a adaptar el O.A.
Precondiciones	1. Carga del O.A.
Flujo Normal	1. Ingreso a entorno web del sistema 2. Seleccionar paquete de O.A. 3. Respuesta a formulario para determinar las mejores configuraciones adaptadas al usuario
Flujo Alternativo	1. El O.A. ya se encuentra cargado al Sistema
Post Condición	Presentación de las recomendaciones a aplicar

Cuadro 5.3 Definición del perfil del estudiante a adaptar el O.A.

Nombre	4. Aplicación de recomendaciones
Actor	Docente
Descripción	El Docente emplea las recomendaciones basadas en las necesidades definidas
Precondiciones	1. Carga del O.A. 2. Respuesta a formulario para determinar necesidades
Flujo Normal	1. Ingreso a entorno web del sistema 2. Seleccionar paquete de O.A. 3. Respuesta a formulario para determinar perfil de usuario
Flujo Alternativo	
Post Condición	Presentación del resultado de aplicar las recomendaciones sobre el O.A.

Cuadro 5.4 Aplicación de la configuraciones planteadas por el Sistema Experto

Nombre	5. Verificar configuraciones aplicadas
Actor	Docente
Descripción	El Docente es el encargado de editar y validar la aplicación de las configuraciones sobre el O.A. cargado para la generación del nuevo O.A.
Precondiciones	1. Aplicación de las recomendaciones dadas por el Sistema Experto
Flujo Normal	1. Ingreso a entorno web del sistema 2. Seleccionar paquete de O.A. 3. Respuesta a formulario para determinar perfil de usuario 4. Se muestran las recomendaciones del S.E al autor para aplicar
Flujo Alternativo	
Post Condición	Edición de las configuraciones aplicadas

Cuadro 5.5 Verificación de las configuraciones aplicadas

Nombre	6. Ensamblaje de Nuevo Objeto de Aprendizaje
Actor	Docente
Descripción	El docente al validar que las recomendaciones se ajustan a las necesidades procede a efectuar el ensamblaje del N.O.A
Precondiciones	1. Aplicación de las recomendaciones dadas por el Sistema Experto 2. Validar la aplicación de las configuraciones dadas por el S.E
Flujo Normal	1. Ingreso a entorno web del sistema 2. Seleccionar paquete de O.A 3. Respuesta a formulario para determinar perfil de usuario 4. Se muestran las recomendaciones del S.E a aplicar al autor 5. Validación y edición de configuraciones aplicadas
Flujo Alternativo	
Post Condición	Publicación electrónica generada

Cuadro 5.6 Ensamblaje de N.O.A.

Nombre	7. Despliegue de N.O.A. sobre LMS
Actor	Docente Estudiante
Descripción	Previo aprobación y edición de configuraciones se procede a realizar pruebas de comprensión lectora del N.O.A.
Precondiciones	1. Aplicación de las recomendaciones dadas por el Sistema Experto 2. Validar la aplicación de las configuraciones dadas por el S.E 3. Ensamblar O.A.
Flujo Normal	1. Ingreso a entorno web del sistema 2. Seleccionar paquete de O.A 3. Respuesta a formulario para determinar perfil de usuario 4. Se muestran las recomendaciones del S.E a aplicar al autor 5. Validación y edición de configuraciones aplicadas
Flujo Alternativo	
Post Condición	Pruebas del O.A. sobre LMS

Cuadro 5.7 Despliegue del N.O.A. sobre un LMS

5.3. Diagramas de Secuencia

Para adaptar el contenido educativo desarrollado y exportado a formato web por el docente en primera instancia se debe indicar al sistema la ruta del O.A. a analizar y procesar, una vez definida dicha ruta, el sistema accede a la estructura D.O.M del O.A. con el objetivo de identificar la información relevante para el procesamiento, en dicha etapa se depuran etiquetas no relevantes y se analizan textos de cada componente Html encontrado en la estructura del O.A. (Ver Figura 5.3).

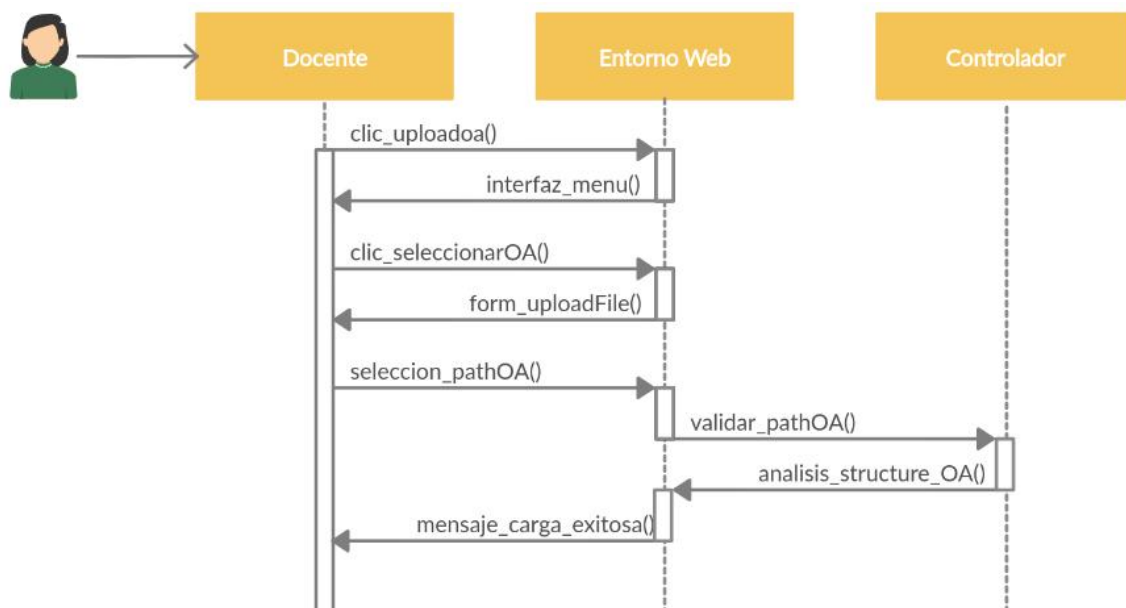


Figura 5.3 Proceso de recomendación de configuraciones basados en las necesidades del alumno

Para determinar dichas recomendaciones el docente interactúa con el sistema experto basado en un formulario de necesidades, dicho formulario refleja en modo de pregunta las configuraciones que se disponen a aplicar sobre los textos analizados y extraídos, mediante el motor de inferencia y la base de conocimientos se realiza un “match” para definir las mejores recomendaciones a aplicarse, reflejándose dichas recomendaciones en una pagina web (Ver Figura 5.4).

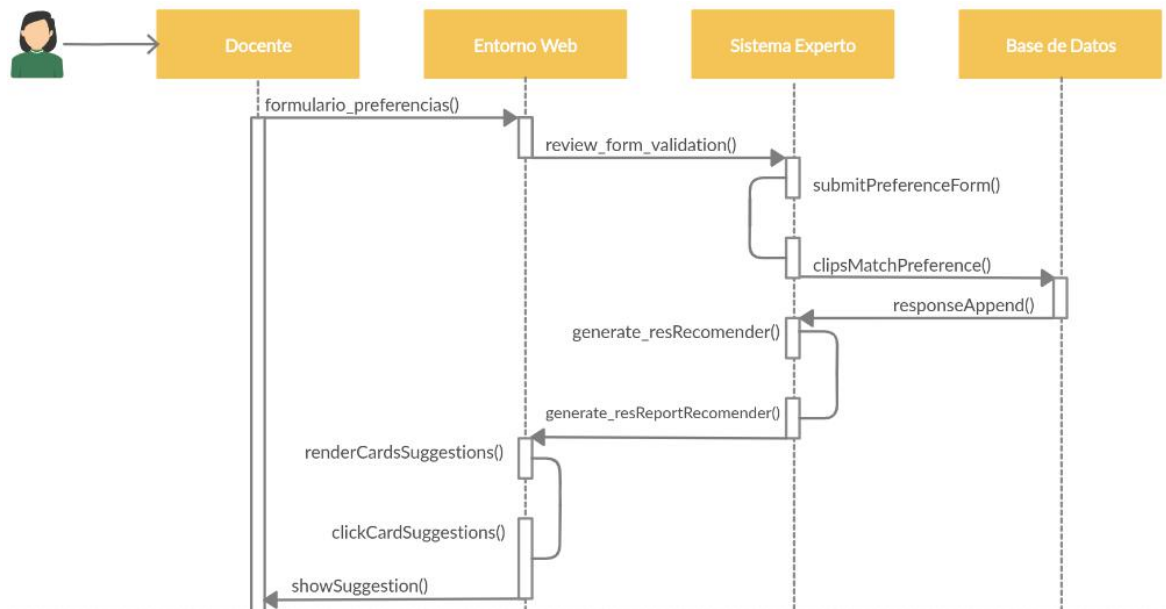


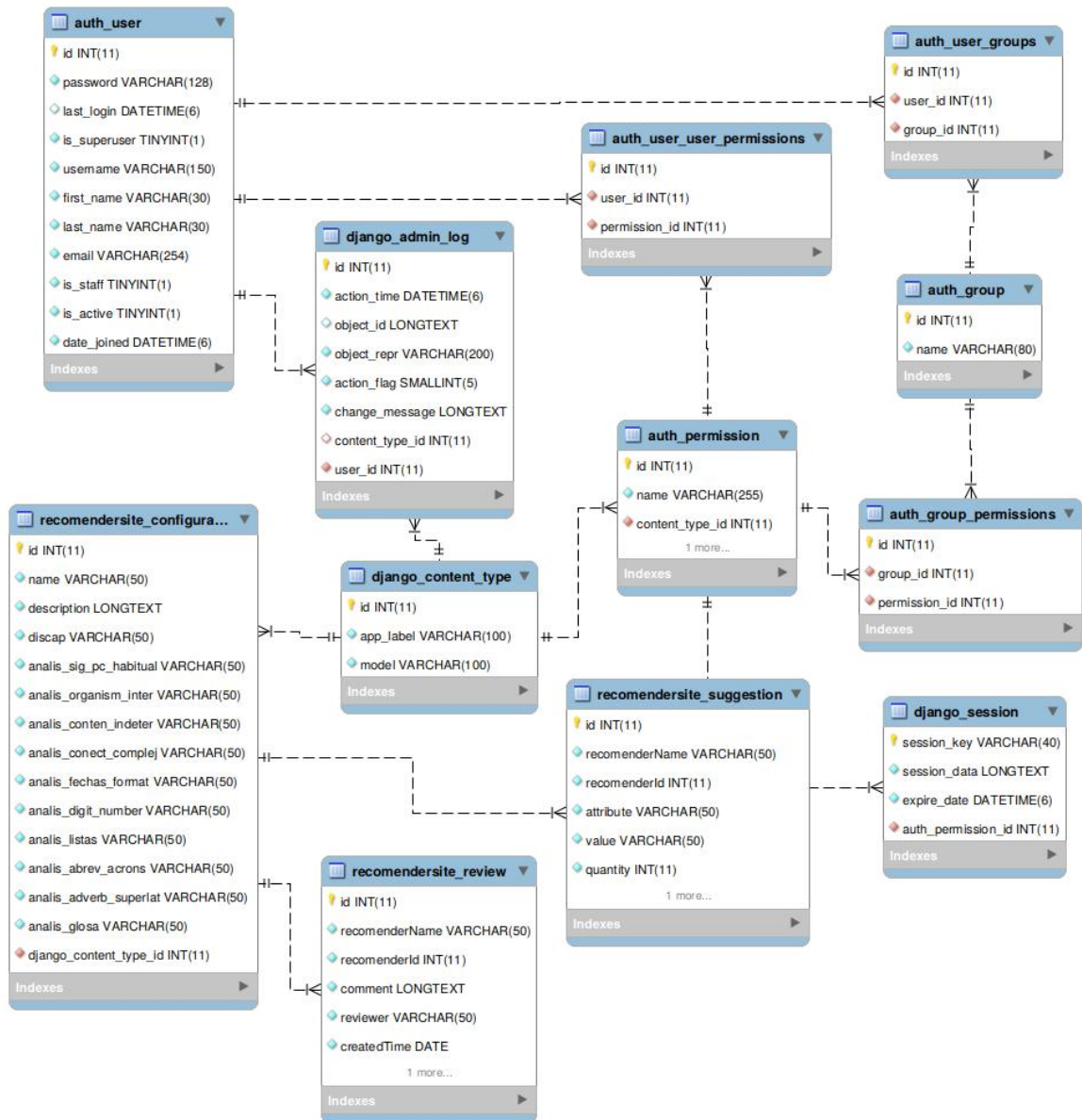
Figura 5.4 Determinación de necesidades para definición del perfil de usuario

5.4. Diagrama objeto relacional del sistema

La estructura de base de datos (B.D) desarrollada permite la interacción entre la base de conocimiento y el motor de inferencia del sistema experto, esto con el objetivo de brindar las recomendaciones de las configuraciones de lectura facil para un perfil de necesidad determinado, gracias a la incorporación de esta B.D se brinda la capacidad a que los expertos en el área de accesibilidad puedan enriquecer la base de hechos del sistema, logrando una autonomía del sistema experto en relación con los desarrolladores.

Como se puede apreciar en la Figura 5.5 dicha estructura se compone de un modelo desarrollado en Django al cual se incorporan nuevos modelos para su funcionalidad como por ejemplo, `recomendersite_configuration`, que es la entidad encargada de almacenar las configuraciones a ser presentadas al docente, en esta se puede evidenciar que contiene el ID de recomendación, su descripción así como el perfil de discapacidad al que beneficia o interviene dicha recomendación, `recomendersite_review`, en esta entidad se identifica a detalle cada una de las configuraciones a ser recomendadas y ser visualizadas en el entorno

web, `recomendersite_suggestion`, dicha entidad auxiliar para el almacenamiento de las recomendaciones definidas para determinados casos de discapacidad, así como la cantidad de veces que esta es recomendada para dicha discapacidad, empleando estas variables para un reporte de métricas técnicas y de desempeño del S.E.



5.5. Arquitectura general de la propuesta planteada

El objetivo de desarrollar una herramienta inteligente para la adaptabilidad de objetos de aprendizaje considerando normativas de accesibilidad enfocadas a personas con discapacidad, es brindar apoyo a los docentes que no estén familiarizados con pautas y estrategias de accesibilidad para la generación de material educativo basado en el perfiles de usuario, a ser impartido en entornos de aprendizaje virtuales, haciendo énfasis en el desarrollo de lectura fácil sobre dicho material educativo.

Para el desarrollo de dicha herramienta es necesario la incorporación de diferentes módulos, así como técnicas inteligentes que permitan identificar y analizar la estructura interna de un O.A. para su posterior manipulación o procesamiento, permitiendo adaptarlo a determinadas especificaciones.

Para lograr este objetivo es necesario que el Docente o Instructor (encargado de impartir el material), proporcione al sistema un O.A. base en el cual se pueda evidenciar:

- Recursos multimedia: imágenes, sonidos, vídeos, entre otros.
- Recursos textuales: información, definiciones, observaciones.

Cabe mencionar que dicha estructura es definida y generada mediante la herramienta eXeLearning, herramienta popular entre los docentes para la creación y publicación de contenidos educativos. A partir de este proceso se obtiene como resultado un N.O.A en formato de libro electrónico (EPUB), brindando la capacidad de ser implementado en conjunto con otras herramientas que permitan aprovechar todo el potencial de dicho material resultante.

Mediante la herramienta planteada, se logra generar un material adaptado a un perfil de usuario, cumpliendo con el objetivo de usabilidad para todos, dicho perfil es identificado en base a criterios de expertos en el área de accesibilidad web y de educación especial.

Además de ello, se brinda un reporte final que permitirá analizar las decisiones que se tomo mediante la herramienta inteligente para apoyar a una retroalimentación dirigida al desarrollador del contenido educativo accesible.

En el desglose de este capitulo se detalla cada una de las técnicas implementadas, las cuales componen los módulos del sistema inteligente, así como las pautas que se adaptaron a un proceso automatizado de adaptabilidad.

Para detallar la funcionalidad el sistema propuesto planteamos la siguiente estructura (Ver Figura 5.6) donde se evidencian las 4 capas desarrolladas, que sirve de guía para comprender su cometido.

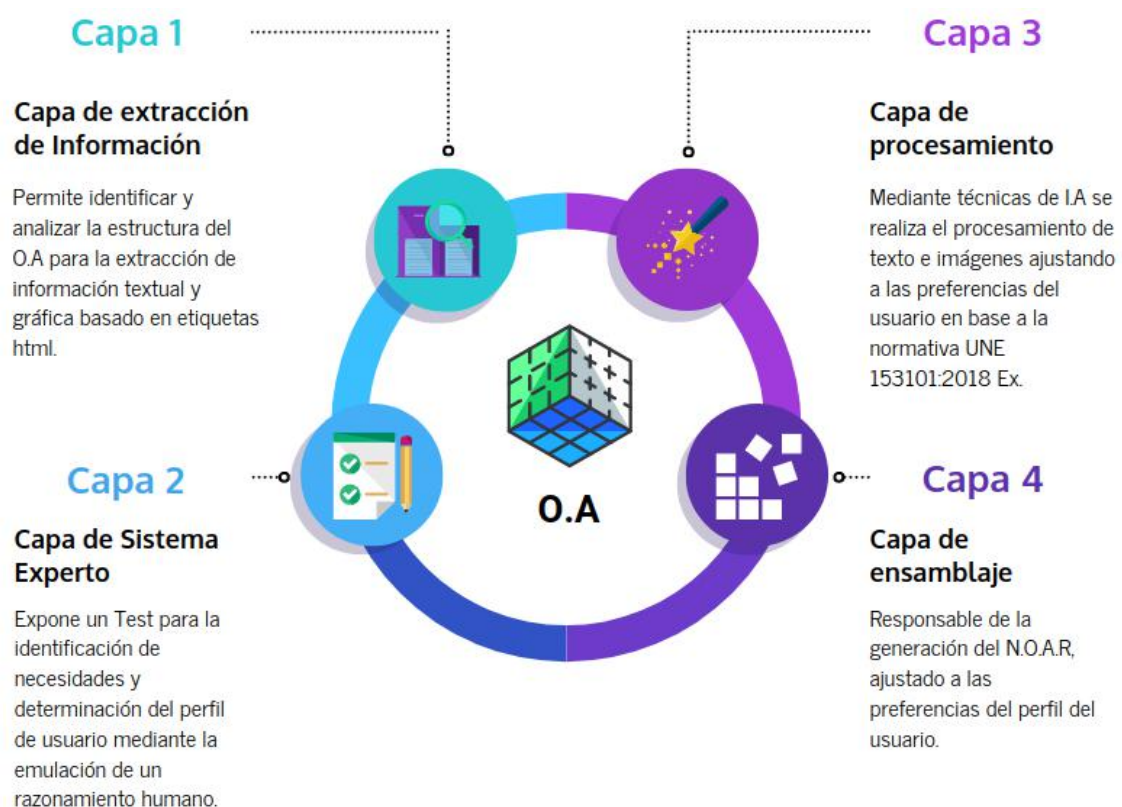


Figura 5.6 Estructura de las capas del sistema inteligente (Fuente: Autor)

5.5.1. Capas del sistema

Para el desarrollo del sistema se proponen 4 capas principales constituidas por sus diferentes módulos detallándose cada una de ellas a continuación:

5.5.1.1. Capa de extracción de información

Esta es una de las partes más importantes del sistema, en esta se desarrollan los métodos para el análisis del O.A., para ello se usan técnicas de Scraping Web empleadas de una manera adecuada y con un propósito idóneo.

Como punto de partida es importante mencionar que el O.A. se lo analiza una vez que lo tenemos como paquete exportado a Web (Ver Figura 5.7), eXeLearning nos permite migrar por esta opción (Archivo ->Exportar ->Sitio Web), teniendo como resultado la siguiente estructura, donde se evidencian los archivos html, css, js, xml, png, jpg, entre otros.

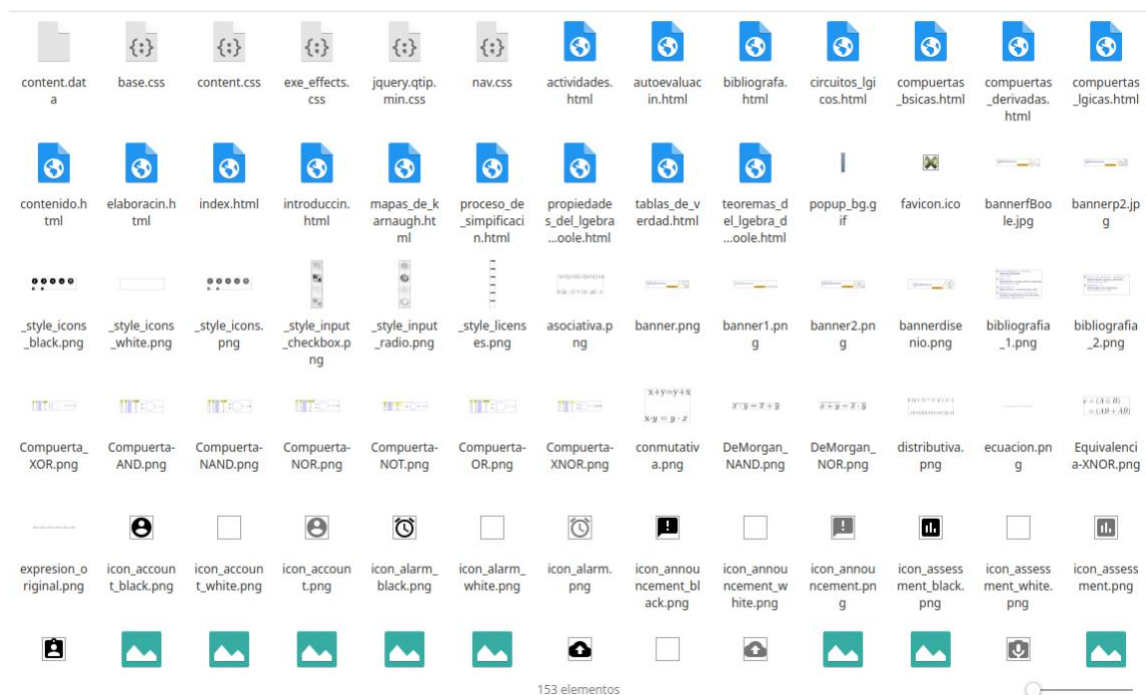


Figura 5.7 Estructura de un O.A exportado en sitio web mediante eXeLearning (Fuente: Autor)

Nota: El sistema fue probado con O.A creados con versiones de eXeLearning >2.2

Una vez que el O.A. es exportado a sitio web, resulta más fácil acceder a los archivos Html o Htm contenedores de la información del O.A y proceder con la extracción de dicha información mediante el uso de analizadores HTML, para posterior tratamiento de información mediante procesamiento de lenguaje natural, para ello con Jsoup accedemos a la estructura DOM de cada una de las páginas individuales. Dicha estructura DOM se muestra en la Figura 5.8.

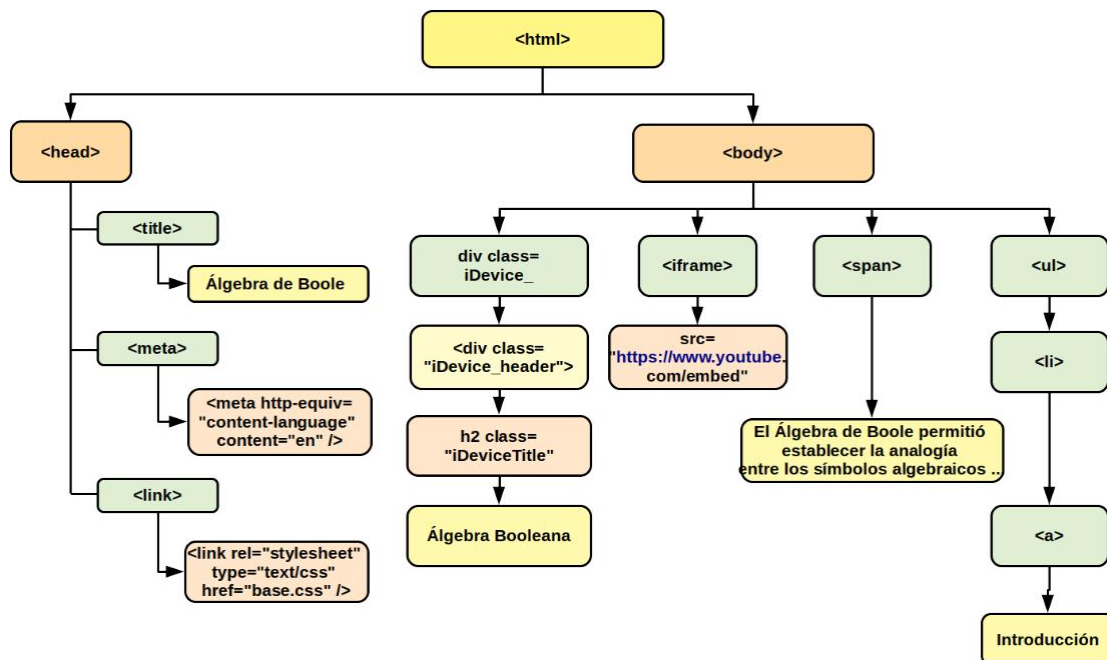


Figura 5.8 Ejemplo de estructura DOM de un recurso .html perteneciente a un O.A desarrollado en eXeLearning (DOM de Figura 4.2) (Fuente: Autor)

Al identificar la estructura sobre la cual se va a proceder a extraer la información se realiza un proceso de depuración de la información en paralelo, proceso por el cual se descartan aquellas etiquetas que no representan o no contienen información relevante para comprensión del lector al presentarle un N.O.A en formato de libro electrónico, entre estas etiquetas se pueden identificar:

- "skipNav"
- "topPagination"
- "siteNav"
- "nodeDecoration"

- "header"
- "bottomPagination"
- "pagination noprt"
- "packageLicense"
- "siteFooter"

En el código 5.1 presentado se puede evidenciar un ejemplo de procesamiento de etiquetas de la arquitectura DOM de un fichero Html que constituye un O.A.

```

1  public String cleanStruct(Document estructura)
2      {estructura.select("title").remove();
3      Element siteNav = estructura.getElementById("siteNav");
4      if(siteNav != null){
5          estructura.getElementById("siteNav").remove();
6      }
7      Element skipNav = estructura.getElementById("skipNav");
8      if(skipNav != null){
9          estructura.getElementById("skipNav").remove();
10         ...

```

Código 5.1 Procesamiento de etiquetas en arquitectura D.O.M

Donde:

```

1  File archivHtml;
2  Document doc = Jsoup.parse(archivHtml, "UTF-8");

```

A más de un proceso de extracción de contenido en etiquetas como "”

```

1  estructura.select("span[style*=display:none]").remove();
2  String text = estructura.text();

```

Al realizar la depuración del contenido de nuestros archivos Html que conforman el O.A. logramos extraer la información textual y que será clave en el proceso de creación de un nuevo O.A. adaptado a Lectura Fácil (Ver Figura 5.9).

```

55 </div>
56 <div id="main-wrapper">
57 <div id="main"><a name="main"></a>
58 <div id="nodeDecoration"><h1 id="nodeTitle">Introducción</h1></div>
59 <div class="iDevice_wrapper preknowledgeiDevice em iDevice" id="id13">
60 <div class="iDevice_emphasis1" >
61 <div class="iDevice_header"><h2 class="iDeviceTitle">Álgebra Booleana</h2></div>
62 <div class="iDevice_inner">
63 <div class="iDevice_content_wrapper">
64 <div id="ta13 55 2" class="block iDevice_content">
65 <p><span style="font-size: medium;">El Álgebra de Boole permitió establecer la analogía entre los símbolos algebraicos y aquellos que representan sus formas
66 <ul>
67 <li><span style="font-size: medium;">El álgebra Booleana o de Boole, consiste en un método para resolver problemas de lógica empleando para ello valores binarios
68 <li><span style="font-size: medium;">El álgebra Booleana se emplea en la construcción de computadoras, circuitos eléctricos, etc.</span></li>
69 <li><span style="font-size: medium;">Es fundamental destacar que los sistemas modernos trabajan a partir de lógica binaria.</span></li>
70 <li><span style="font-size: medium;">En términos generales, el álgebra de Boole se puede emplear a cualquier sistema en que cada variable posea dos estados
71 <li><span style="font-size: medium;">Las computadoras modernas realizan sus operaciones y almacenan la información empleando valores binarios (1 y 0) y real
72 <li><span style="font-size: medium;">Se emplea en el diseño de circuitos electrónicos.</span></li>
73 <li><span style="font-size: medium;">Se emplea ampliamente en la programación (condiciones lógicas).</span></li>
74 </ul>
75 <p><span style="font-size: medium;">Te invitamos a ver este video, elaborado por RTE-Ireland's National Public Service Media, que presenta la importancia de
76 <p><br /> <iframe width="560" height="315" style="display: block; margin-left: auto; margin-right: auto;" src="https://www.youtube.com/embed/aEjzJLV-YJI" fr
77 </p>
78 </div>
79 </div>
80 </div>
81 </div>
82 </div>
83 </div>
84 </div>
85 </div>
86 </div>
87 </div>
88 </div>
89 </div>
90 </div>
91 </div>
92 </div>
93 </div>
94 </div>
95 </div>
96 </div>
97 </div>
98 </div>
99 </div>
100 </div>

```

Figura 5.9 Resultado de scraping web en archivos html del O.A. (Fuente: Autor).

Se debe indicar que este proceso de análisis y procesamiento de estructuras se lo realiza para cada #N archivos Html encontrados dentro del O.A. y de manera automatizada por el sistema inteligente, el resultante de dicho análisis es almacenado en un fichero de texto para su posterior uso en el PLN (Procesamiento de Lenguaje Natural). Para relacionar cada file correspondiente a cada estructura D.O.M analizada se procede a identificar la etiqueta “<title></title>” dentro de esta estructura y procesar dicho nombre para convertirlo en un identificador intuitivo correspondiente para cada archivo de texto resultante: En el código 5.2 se puede evidenciar un ejemplo de procesamiento de identificadores para cada file resultante del scraping web.

```

1 public static String cleanString(String texto) {
2     texto = Normalizer.normalize(texto, Normalizer.Form.NFD);
3     texto = texto.replaceAll(
4     [\\p{InCombiningDiacriticalMarks}]', '');

```

```
5 return texto; }
```

Código 5.2 Procesamiento de identificadores de scraping web

Donde

```
1 texto = '<title>title__</title>'
```

5.5.1.2. Capa de sistema experto

Dentro de los objetivos planteados con este desarrollo se encuentra la adaptabilidad de objetos de aprendizaje basados en los perfiles de usuario, esto quiere decir que cada O.A. resultante, es personalizado para determinado perfil de discapacidad, y para el desarrollo de esta propuesta se han seleccionado las siguientes discapacidades: Discapacidad Auditiva (Sordera - prelingüística), Discapacidad Cognitiva, Adulto Mayor, Bajo nivel de alfabetización.

Escogidos por criterios de necesidad de acceso a información escrita, comprensión lectora y soporte en procesos iniciales de lectoescritura.

Para lograr este objetivo se plantea emular un razonamiento humano con el apoyo de Clips, que permita determinar en base a las necesidades y preferencias de cada usuario al cual esta pensado adaptar el contenido, las mejores practicas a ser adoptadas con el fin de potenciar un O.A. en el marco de la Lectura Fácil.

Mediante el uso de Clips se genera una capa inteligente que permite brindar ciertas recomendaciones a partir de la identificación de necesidades resultantes de aplicar un formulario dirigido al Docente, creador y responsable de adaptar el N.O.A., en este formulario se presentan preguntas y respuestas concretas que permiten deducir cuales son las exigencias a cumplir y a las cuales el sistema experto responderá con tipos de configuraciones o practicas apegadas a la normativa de Lectura fácil UNE:153101 sobre la adaptabilidad del texto informativo en los O.A., que se deberán emplear para cubrir la necesidad de determinado usuario.

Con el uso de herramientas como Django, Clips, Pyclips y Mysql se propone un esquema lógico del S.E desarrollado (Ver Figura 5.10).

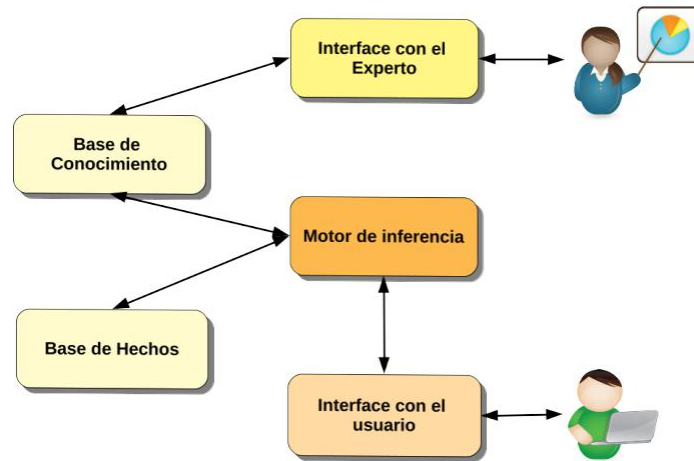


Figura 5.10 Arquitectura básica del sistema experto desarrollado.

5.5.1.2.1. Definición del sistema de gestión: se plantea un entorno web enlazado al núcleo de procesamiento heurístico, el cual brinda a los docentes la facilidad de adaptar los O.A. a determinados perfiles de usuarios, proponiendo la siguiente estructura general del sistema de gestión:

```

recomenderexp/
├── recomender
│   ├── __init__.py
│   ├── pycache
│   ├── __init__.cpython-35.pyc
│   ├── settings.cpython-35.pyc
│   ├── urls.cpython-35.pyc
│   └── wsgi.cpython-35.pyc
│   ├── settings.py
│   ├── urls.py
│   └── wsgi.py
├── recomendersite
│   ├── admin.py
│   ├── apps.py
│   ├── __init__.py
│   ├── migrations
│   │   └── __init__.py
│   ├── models.py
│   ├── tests.py
│   ├── urls.py
│   └── views.py
├── static
│   ├── assets
│   ├── css
│   ├── images
│   └── js
├── templates
│   ├── acercade.html
│   ├── base.html
│   ├── index.html
│   ├── preferences.html
│   └── nuevarec.html
└── manage.py
    
```

Figura 5.11 Arquitectura general del sistema definido para el S.E desarrollado.

Modelos: para definir las tablas y las relaciones entre estas en la Base de Datos recurrimos al empleo de los modelos, en los cuales se define el nombre, a si como sus atributos y relaciones entre estas. Gracias a la arquitectura que se modelo en Django, los procesos de ingreso, modificación y eliminación (CRUD) se los realiza de una manera óptima, empleados en procesos administrativos y los cuáles son presentados a los usuarios finales en un entorno web al ser invocados (Ver código 5.3).

```
1 # Creacion de modelo configuraciones.
2 class configuracion(models.Model):
3     name = models.CharField(max_length=50)
4     description = models.TextField()
5     discap = models.CharField(max_length=50)
6     analisis_sig_pc_habitual = models.CharField(max_length=50)
7     analisis_organism_inter = models.CharField(max_length=50)
8     analisis_conten_indeter = models.CharField(max_length=50)
9     analisis_conect_complej = models.CharField(max_length=50)
10    analisis_fechas_format = models.CharField(max_length=50)
11    analisis_digit_number = models.CharField(max_length=50)
12    analisis_listas = models.CharField(max_length=50)
13    analisis_abrev_acrons = models.CharField(max_length=50)
14    analisis_adverb_superlat = models.CharField(max_length=50)
15    analisis_glosa = models.CharField(max_length=50)
```

Código 5.3 Ejemplo de modelo configuraciones en Django

Administración: mediante la definición del componente de administración (Ver código 5.4) se puede acceder a la gestión (Edición, Visualización, Eliminación) de los modelos definidos, en un entorno web.

```
1 # -*- coding: utf-8 -*-
```

```
2 from __future__ import unicode_literals
3 from django.contrib import admin
4 from recomendersite.models import Configuraciones
5 from recomendersite.models import Review
6 from recomendersite.models import Suggestion
7 admin.site.register(Configuraciones)
8 admin.site.register(Review)
9 admin.site.register(Suggestion)
```

Código 5.4 Ejemplo de registro de modelos en entorno administrativo Django

Vista: por medio de la implementación de vistas se genera una comunicación mediante los objetos de solicitud `HttpRequest` con metadatos de solicitud entre las plantillas definidas, que son presentadas al usuario final y los modelos de datos que se crearon. Cabe indicar que cada vista devuelve un objeto `HttpResponse` y `JsonResponse`. Por medio de la vistas se desarrolla la lógica para procesar todas las solicitudes Web y proyectar las respuestas de estas peticiones a los usuarios finales (docente) (Ver código 5.5). Gracias a esta comunicación presentamos las recomendaciones generadas por el sistema experto, así como el formulario para la inserción de nuevas recomendaciones a la base de hechos, que posteriormente serán presentados al docente.

```
1 def generapreferencias(request):
2     result= clipsMatchPreference(request.POST)
3     response = []
4     responseReport = []
5     if result != None:
6         for val in result.split('---'):
7             if "," in val:
8                 val2 = val.split(',')
```

```

9         configuracion = Config.objects.get(id=int(val2[0]))
10        response.append({"id": configuracion.id})
11        responseReport.append({"id": configuracion.id, "name":
12        configuracion.name, "pauta": val2[1]})
13        # ESCRITURA DEL RESULTADO DE LA RECOMENCION PARA POSTERIOR
14        ANALISIS DE LENGUAJE
15        file1 = open(settings.REPORT_DIR+ "/resRecomender.txt", "w")
16        file1.writelines(str(response) for response in response)
17        file1.close() #to change file access modes
18        file2 = open(settings.REPORT_DIR+ "/resReportRecomender.txt", "w")
19        file2.writelines(str(report) for report in responseReport)
20        file2.close()
21        file1 = open(settings.REPORT_DIR+ "/resRecomender.txt", "r+")
22        return JsonResponse(responseReport, safe=False)

```

Código 5.5 Ejemplo respuesta de recomendaciones basadas en resultado de reglas.

URLs: para que cada página web sea desplegada en base a los modelos de vistas definidos se necesita una llamada mediante la especificación de una url que sirva de comunicación, para representar estos enlaces nos dirigimos al archivo urls y los añadimos (Ver código 5.6).

```

1 from django.conf.urls import url
2 from . import views
3 urlpatterns = [
4     url(r'^$', views.index, name='index'),
5     url(r'^preferences/$', views.preferencePage,
6     name='preferencePage'),
7     url(r'^new/$', views.newConfigurationPage, name='
8     newConfigurationPage'),
9     url(r'^about/$', views.aboutUsPage, name='about
10    UsPage'),

```

```
url(
    '^api/reviews$',
    views.ProcessReviews,
    name='process_reviews'),
```

```

9     url(r'^api/createConfiguration', views.createConfiguration
10    on, name='createConfiguration'),
11    url(r'^api/generapreferencias', views.generapreferencias,
12    name='generapreferencias'),
13 ]

```

Código 5.6 Especificación de URL's en Django.

Templates: para que el usuario final interactúe con el S.E se definen los templates que son básicamente archivos html (Ver código 5.7). Se definen templates para el formulario de preferencias, creación de nuevas configuraciones, un template base para el despliegue del formulario de preferencias y un landing page de presentación del S.E.

```

1 {% extends "base.html" %}
2 {% load staticfiles %}
3 {% block css-files %}
4 {% endblock %} {% block content %}
5 <h1>Ortotipografía</h1>
6 <div role="tabpanel" class="tab-pane active" id="discover">
7 <div class="design-process-content">
8 <h3 class="semi-bold">Pautas y recomendaciones relacionadas con la
9 ortotipografía:</h3>
10 <div>
11 <label class="four wide">1. P1. Desea que en su
12 texto se analicen signos poco habituales como %, &,
13 etc. ¿?</label>
14 <div class="inline fields">
15 <div class="three wide field">
16 <div class="ui radio checkbox">

```

```

14         <inputrequired style="margin-left: 30px;" type="radio" name=
"analys_sig_pc_habitual" tabindex="0" value="TRUE" class="hidden
"><label>Si</label></div>
15     <div class="three wide field">
16         <div class="ui radio checkbox">
17             <inputrequired type="radio" name="
analys_sig_pc_habitual" tabindex="0" value="FALSE"
class="hidden">
18             <label>No</label>
19         </div>
20     </div> </div> </div> <div>

```

Código 5.7 Ejmplo de Template preferencias en Django.

Files - CLIPS: mediante la construcción de deftemplates desarrollamos hechos con planti-llas siendo estos un poco más complejos, los cuales se emplearan a posterior en las reglas del S.E (Ver código 5.8), mediante la implementación de una serie de slots podemos almacenar datos relacionados a un hecho definido, gracias a esta propiedad podemos ingresar valores por defecto y en un orden dado, permitiendo de esta manera que el usuario ingrese o no todos los campos para este hecho.

```

1 (deftemplate
2   (slot ID (type NUMBER))
3   (slot name (type STRING))
4   (slot discap (type STRING))
5   (slot analys_sig_pc_habitual (type STRING))
6   (slot analys_organism_inter (type STRING))
7   (slot analys_conten_indeter (type STRING))
8   (slot analys_conect_complej (type STRING))
9   (slot analys_fechas_format (type STRING))
10  (slot analys_digit_number (type STRING))

```

```

11 (slot analisis_listas (type STRING))
12 (slot analisis_abrev_acrons (type STRING))
13 (slot analisis_adverb_superlat (type STRING))
14 (slot analisis_glosa (type STRING))
15 )

```

Código 5.8 Ejemplo de Template en Clips para la representación de una recomendación o configuración.

5.5.1.2.2. Base de hechos: en esta base de hechos o también conocida como base de datos global se registran y se inicializan los hechos. Para la definición de esta base se elabora una forma de introducción de hechos por conjunto y de forma remota o diferida, para ello mediante la instrucción (deffacts) aprovechamos la ventaja de que los hechos no se los carga directamente en memoria, si no estos se cargan cuando el sistema se reinicia mediante (reset) (Ver código 5.9).

```

1 (deffacts configuraciones
2 (recomender (ID 78) (name "No se debe utilizar el punto y coma (;)")
3 (discap "discCogni") (analisis_sig_pc_habitual "NO")
4 (analisis_organism_inter "NO")
5 (analisis_conten_indeter "NO")
6 (analisis_conect_complej "NO")
7 (analisis_fechas_format "YES")
8 (analisis_digit_number "YES")
9 (analisis_listas "YES") (analisis_abrev_acrons "YES")
10 (analisis_adverb_superlat "YES")
11 (analisis_glosa "NO"))
12 ;; definicion de hechos de perfil de discapacidad auditiva
13 (recomender (ID 39)
14 (name "Se deberia evitar el uso de los numeros romanos. En el caso

```


	Auditiva	Cognitiva	Otros	
	Sordera pre-ligüística	Disc. intelectual	Bajo nivel alfabetización	Adulto Mayor
6.1 Pautas y recomendaciones relacionadas con la ortotipografía				
1 No se deben escribir palabras ni frases con todas sus letras en mayúsculas, excepto cuando se trate de siglas.		X	X	
2 Se debe utilizar la mayúscula inicial al principio de un párrafo o un título, después de punto o en nombres propios.		X	X	
3 Debería sustituirse el punto y seguido por un punto y aparte o por una conjunción.	X	X	X	X
4 Se deberían utilizar los dos puntos (:) cuando se introduce una lista que enumera más de tres elementos.		X		
5 No se debe utilizar el punto y coma (;).	X	X	X	X
6 Se debería evitar el uso de paréntesis, corchetes y signos ortográficos poco habituales (%, &, /, . . . , etc.).	X	X	X	
7 No se debe utilizar etcétera, ni los puntos suspensivos (. . .). Se puede sustituir por entre otros, y muchos más y otras frases similares.		X	X	
8 No se deberían usar las comillas (las españolas « » o las inglesas “ ”). Cuando se utilicen, como en el caso de las citas textuales deben ir acompañadas de una explicación.	X	X	X	
6.2 Pautas y recomendaciones relacionadas con el vocabulario y las expresiones				
9 Se debería elaborar un glosario y/o incluir glosas en el documento, si hay varios términos difíciles de entender.	X	X	X	X
10 Se deben evitar los adverbios terminados en -mente.		X		
11 Se deben evitar los superlativos. Es recomendable añadir el adverbio muy al adjetivo o al adverbio.	X	X	X	
12 Se deben evitar las abreviaturas.	X	X	X	X
13 Se debería evitar el uso de siglas.	X			

	Auditiva	Cognitiva	Otros	
	Sordera pre-ligüística	Disc. intelectual	Bajo nivel alfabetización	Adulto Mayor
14 Se pueden utilizar acrónimos cuando su uso esté extendido en el idioma. No obstante, la primera vez que se utilicen se debería explicar su significado.	X	X	X	X
15 Se debería evitar el uso de palabras de contenido indeterminado como: cosa, algo o asunto.	X	X	X	X
16 Se deberían escribir los números en cifra. Los números con muchos dígitos son difíciles de leer y escribirlos con letra puede facilitar su lectura.	X	X	X	
17 Los números de teléfono se deberían separar por bloques.	X	X		X
18 Se debería evitar el uso de números ordinales (1.º) y sustituirlos por números cardinales (1).	X	X	X	
19 Se debería evitar el uso de fracciones y de porcentajes. Si es necesario incluirlas se debería describir su equivalencia o explicación.	X	X	X	X
20 Se debería evitar escribir la fecha con guiones o barras. Se debería escribir la fecha completa y el nombre del día cuando aporte información adicional para la comprensión del texto.		X	X	X
21 Se debería evitar escribir la hora en formato 24 horas.	X	X	X	
22 Se debería evitar el uso de los números romanos. En el caso de siglos, reyes y papas se recomienda poner el número romano e indicar cómo se lee.	X	X	X	
6.3 Pautas y recomendaciones relacionadas con frases y oraciones				
23 Se debería evitar el uso de conectores complejos entre oraciones, como por lo tanto, no obstante, por consiguiente o sin embargo.	X	X	X	
6.4 Pautas y recomendaciones relacionadas con la organización del texto y el estilo				
24 Si se van a enumerar los elementos relacionados con una idea en una sola frase, se debería utilizar la coma para separarlos. Siempre que sea posible, si hay varios elementos (como una lista de tareas), se debe utilizar un listado.	X	X	X	X

	Auditiva	Cognitiva	Otros	
	Sordera pre- ligüística	Disc. intelectual	Bajo nivel alfabe- tización	Adulto Mayor
25 Se debería escribir cada capítulo o subdivisión del texto relativo a una materia o asunto en una página nueva.	X	X	X	X
26 Se debería mantener una longitud de línea homogénea, siempre que sea posible.	X	X	X	X
27 Se debería usar un interlineado mínimo de 1,5 adecuándolo al tamaño de la tipografía y al tamaño del soporte.	X	X	X	X
28 Se debería alinear el texto a la izquierda.	X	X		X
29 Se debe evitar la justificación de textos.	X	X	X	X
30 Se debe usar un contraste adecuado entre el color de fondo de la página y el color de los caracteres.		X		
31 El fondo del texto debe ser de color liso.		X		
32 El uso de la negrita debería limitarse para destacar las palabras que se explican en la glosa, en el glosario o en los complementos paratextuales.	X	X	X	X
33 Se debe evitar el uso del estilo cursiva, subrayado, sombreado, contorno o relieve.		X	X	X
34 No se deben usar marcas de agua.		X		
35 Cuando en un documento se incluyan textos en Lectura Fácil y textos no escritos en Lectura Fácil, se debe advertir al lector qué partes están escritas en Lectura Fácil y cuáles no.	X	X	X	X

Cuadro 5.8 Definición de pautas y recomendación basadas en la normativa UNE:153101 que conforman la base de conocimiento del Sistema Experto

```
15 de siglos, reyes y papas se recomienda poner el numero romano e
16 indicar como se lee.")
17 (discap "discAudi") (analisis_sig_pc_habitual "NO")
18 (analisis_organism_inter "YES")
19 (analisis_conten_indeter "YES")
20 (analisis_conect_complej "YES")
21 (analisis_fechas_format "YES")
22 (analisis_digit_number "YES")
23 (analisis_listas "YES") (analisis_abrev_acrons "YES")
24 (analisis_adverb_superlat "YES") (analisis_glosa "YES"))
```

Código 5.9 Ejemplo de introducción remota de hechos

Al definir nuestros hechos los estructuramos en base a la cantidad de preguntas y respuestas planteadas para un perfil de usuario determinado, en el cual se evidencia el ID de pauta a recomendarse, así como el perfil de discapacidad para determinados casos.

5.5.1.2.3. Base de conocimientos: la base de conocimientos en combinación con el motor de inferencia son la parte más importante del S.E, en esta B.C se representa el conocimiento del experto definido anteriormente en el cuadro 5.8 por medio de un conjunto de reglas con un significado propio de tipo declarativo (una descripción del conocimiento), e inferir adecuadamente las reglas que se deben ejecutar para recomendar las mejores configuraciones basadas en las necesidades planteadas por medio del motor de inferencias.

- Definición de reglas: el core del sistema experto se basa en un conjunto de reglas que determinan las configuraciones que mejor se ajustan a su perfil tomando de referencia las necesidades del usuario. Para cumplir este objetivo se definen dos estructuras relevantes.

- Reglas para determinación de preferencias.

- Reglas para determinación de descripción de preferencias.

Por medio de estas brindamos al docente las mejores recomendaciones que se ajustan a la necesidad de sus estudiantes y cada una con una breve descripción de usabilidad del mismo a partir de lo que nos indica la normativa UNE:153101.

- Reglas para determinación de descripción de preferencias: cabe indicar que estas reglas se ejecutan en una prominencia = 2, esto quiere decir que en primera instancia se realiza un proceso de identificación de descripciones (review) de las configuraciones a inferir, teniendo en cuenta que estas son recuperadas de la B.D (Base de datos) mediante el api de abstracción de B.D proporcionada por Django permitiendo construir un QuerySet gracia al Manager del modelo definido, en este caso Review, todo esto en nuestro view.py.

Cabe señalar que se debe realizar un proceso de importación de modelos para que esto surja efecto como se evidencia en el código 5.10:

```
1 from recommendersite.models import Configuraciones,  
    Review, Suggestion  
  
1 reviews = Review.objects.filter(recomenderId=int(request.  
    st.GET['recomenderId'])).order_by("-id")
```

Código 5.10 Recuperación de datos de Review de una recomendación aplicando un filtro de parámetros.

Y para determinar qué descripción se debe obtener para cada recomendación se emplean unas reglas de Clips, que en base al ID de recomendación generada se infiere dicha descripción, como se muestra en el código 5.11.

```

1 (defrule review-analis-sig-pc-habitual
2   (declare (salience 2))
3
4   ?d<- (recomender (ID ?id)
5         (analis_sig_pc_habitual ?origin))
6
7         (suggestion (recomender-id ?id)
8                   (attribute "analis_sig_pc_habitual")))
9
10        (not (suggestion (recomender-id ?id)
11                        (attribute "analis_sig_pc_habitual")))
12 => (modify ?d                                     (analis_sig_pc_habitu
13 al ?value)))

```

Código 5.11 Regla para determinación de review para una recomendación.

Cabe señalar que existe la posibilidad de generar una base de hechos (suggestion) similar a la planteada anteriormente en la cual se puede registrar por parte del docente cada review correspondiente a una configuración determinada.

Este proceso se lo realiza para cada configuración teniendo un total de 10 reglas de descripciones.

- Reglas para determinación de preferencias: a la par que se determinan las descripciones se realiza el proceso de obtener las recomendaciones de estas descripciones, siendo estas las que se ajustan a las necesidades planteadas por el docente. Gracias a la plantilla de hechos resulta más sencillo definir las reglas para determinar qué configuración o configuraciones satisfacen la condiciones. Es por ello que con una prominencia = 1 se desarrollan las reglas que permiten realizar un “matching” entre las configuraciones y preferencias dadas, se puede evidenciar una de estas en el código 5.12.

```

1 (defrule recomender-matching
2   (declare (salience 1))
3
4   (recomend
5     er (ID ?ID)

```

```

5      (name ?name)(discap          ?discapacidad)(
      analis_sig_pc_habitual          ?sig_pc_habitual)
      (
      analis_organism_inter          ?organism_inter)
      (
      analis_conten_indeter          ?conten_indeter)
6      (
7      analis_conect_complej          ?conect_complej)
      (analis_fechas_format ?fechas_format)
8      (analis_digit_number ?digit_number)
9      (analis_listas ?listas)
10     (analis_abrev_acrons ?abrev_acrons)
11     (analis_adverb_superlat ?adverb_superlat)
12     (analis_glosa ?glosa))
13 (preference (discap "?"|?discapacidad)
14             (analis_sig_pc_habitual "?"|?sig_pc_habitual)
15             (analis_organism_inter "?"|?organism_inter)
16             (analis_conten_indeter "?"|?conten_indeter)
17             (analis_conect_complej "?"|?conect_complej)
18             (analis_fechas_format "?"|?fechas_format)
19             (analis_digit_number "?"|?digit_number)
20             (analis_listas "?"|?listas)
21             (analis_abrev_acrons "?"|?abrev_acrons)
22             (analis_adverb_superlat "?"|?adverb_superlat)
23             (analis_glosa "?"|?glosa))
24=>
25 (printout t crlf "---Recomendacion SIO EXPERT---" crlf)
26 (printout t crlf ?ID          "," ?name          "," ?discapacidad
      "," ?sig_pc_habitual "," ?organism_inter "," ?
      conten_indeter "," ?conect_complej "," ?fechas_format  ","

```

```
?digit_number "," ?listas "," ?abrev_acrons "," ?  
adverb_superlat "," ?glosa "----" crlf))
```

Código 5.12 Regla para determinación de configuraciones basadas en un perfil de necesidades.

5.5.1.2.4. Motor de inferencias: la función principal del motor de inferencias es seleccionar las reglas que satisfacen la solución a un problema determinado, y al encontrarlas informa de la mejor encontrada, para determinar dicha solución se emplean un programa de control desarrollado en Django de propósito general que realiza una exploración metódica de la base de conocimientos. En contexto de esta propuesta nos vamos a centrar en dos procesos claves de este programa de control, el primero de ellos es un conjunto de script desarrollado en JavaScript, y el segundo en los métodos de nuestro `view.py`, los dos trabajan en conjunto sobre la arquitectura de Django propuesta en [la definición del sistema de gestión](#). Con el objetivo de interpretar la funcionalidad de nuestro motor de inferencia planteamos el diagrama que se observa en la Figura 5.12.

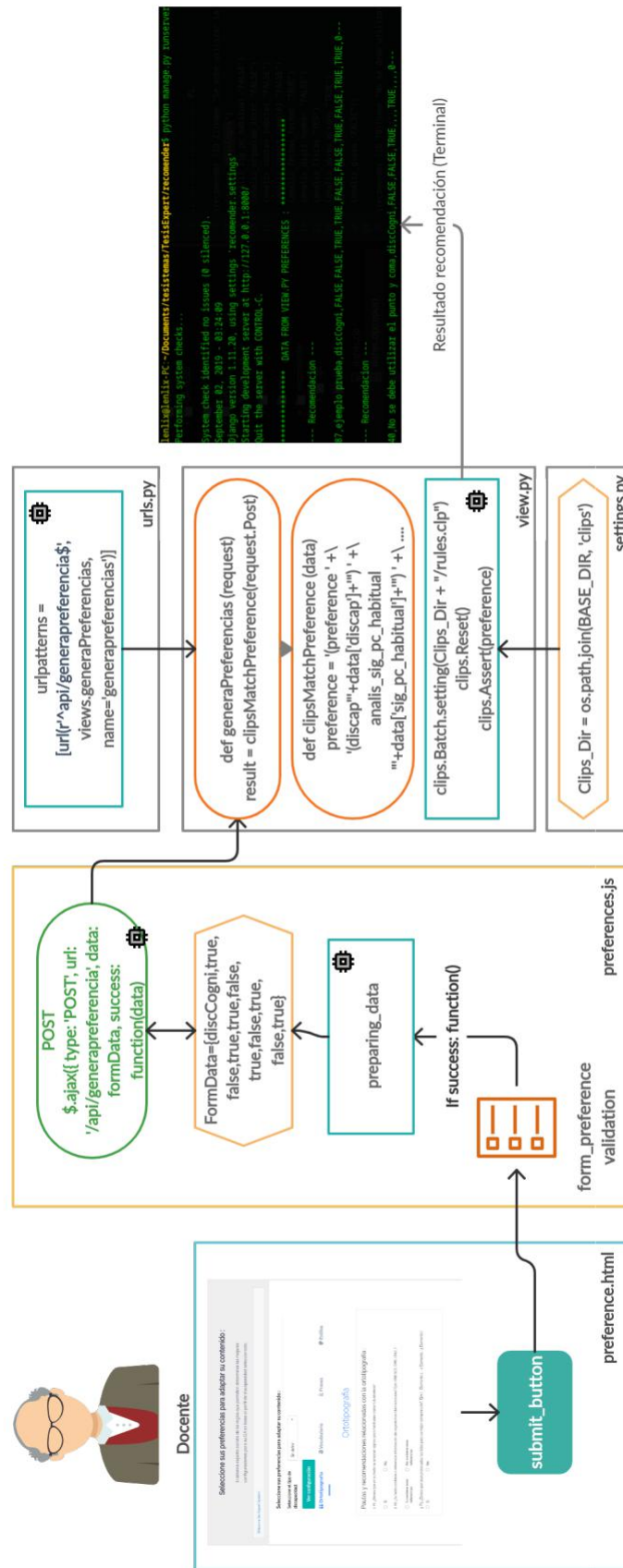


Figura 5.12 Arquitectura general del motor de inferencia empleado en el sistema experto.

Gracias al diagrama planteado podemos identificar el empleo de servicios ajax para la comunicación entre la base de conocimiento y la interface con el usuario (docente). Mediante la abstracción de datos a partir de la vista `preference.html` invocamos a un método `form_preferencevalidation()`, en el cual se valida cuales de los input de formulario son ingresados y cuales no, dicho método es invocado mediante `submit_button` de la interfaz `.html`, si la validación es correcta se procede a preparar los datos para generar el arreglo temporal `formData` constituido por el valor de los `checked` ingresados. Una vez definido este arreglo se procede a generar la solicitud POST al `api/generapreferencia`, definido en `view.py`, y como proceso final mediante `clipsMatchPreference` generamos nuestro asset `preference` con el cual se ejecuta la regla `recomender-matching`, antes definida.

5.5.1.2.5. Interface con el Experto: este módulo es el encargado de presentar al experto del área de accesibilidad en entornos educativos una interfaz (Ver Figura 5.13) en la que puede consultar e incluir nuevos conocimientos al sistema experto sin la intervención del desarrollador del sistema. Para ello se realiza un proceso similar descrito en el M.I con la diferencia de que se abstraen los valores de `newConfigurationPage.html` y por medio del método `inserthecho(data)` en `view.py` se ingresa dicho conocimiento en la BD, como se puede evidenciar en el código 5.13.

```
1 recomender=Configuraciones(name=data['name'],descripti
   on=data['description'],
2
   discap=data['discap'],analisis_sig_pc_habitu
3 al=data['analisis_sig_pc_habitual'],
   analisis_organism_inter=data['analisis_organism_inter
4 '],analisis_conten_indeter=data['analisis_conten_indeter'],
   analisis_conect_complej=data['analisis_conect_comple
   j'],analisis_fechas_format=data['analisis_fechas_format'],
```

```

5      analis_digit_number=data['analis_digit_
6      number'],analis_listas=data['analis_listas'],
7      analis_abrev_acrons=data['analis_abrev_acrons'],ana
lis_adverb_superlat=data['analis_adverb_superlat'],
      analis_glosa=data['analis_glosa']) recomender.save()

```

Código 5.13 Método para ingreso de nuevos hechos en la Base de Datos.

Figura 5.13 Interface con el experto para el ingreso de hechos remotos

5.5.1.2.6. Interface con el usuario: este entorno se presenta al docente encargado de adaptar el O.A. a la normativa UNE:153101, definido como preference.html y enlazado con el motor de inferencias por medio de preference.js, explicado anteriormente.

Resultado de este esquema se lo refleja como resultado en las interfaces presentadas al docente (Figuras 5.14, 5.15 y 5.16).

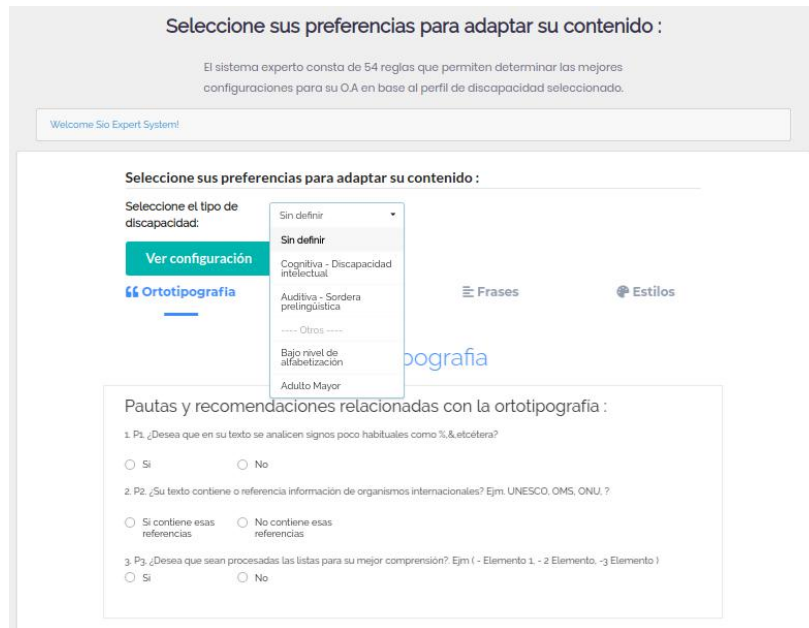


Figura 5.14 Interface con el usuario (Docente) donde se presenta el formulario de necesidades.

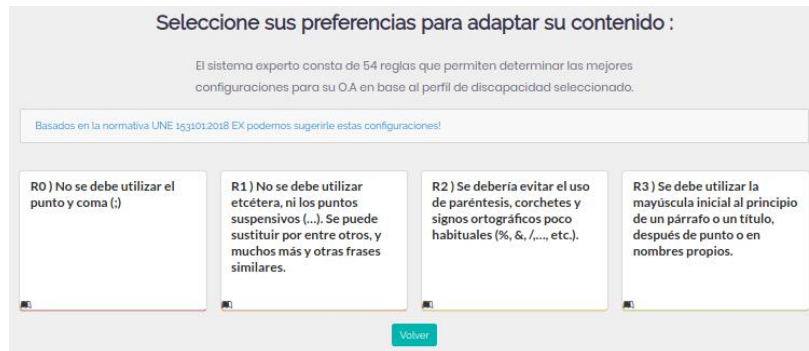


Figura 5.15 Interface con el usuario (Docente) donde se presenta las recomendaciones dadas por el S.E.



Figura 5.16 Interface con el usuario (Docente) donde se presenta la descripción de la recomendación.

5.5.1.3. Capa de procesamiento:

En esta capa se aplican las pautas y configuraciones de la normativa UNE:153101 a los textos que se obtuvieron en la capa de extracción de contenidos.

Como ya se definió anteriormente, por medio del S.E logramos identificar los ID de las recomendaciones que cumplen con las restricciones planteadas por el docente para adaptar el O.A, dicho resultado es almacenado en ficheros para ser analizado por el módulo de procesamiento de lenguaje natural desarrollado en python, en el cual con el uso de spaCy y expresiones regulares, se logra procesar los textos respetando las recomendaciones dadas.

A continuación procedemos a describir las fases que componen esta capa de procesa-miento:

- Como primera fase identificamos y accedemos a los archivos resultados de la capa de extracción de información, siendo estos, los que se muestran en la Figura 5.17.



Figura 5.17 Ficheros resultados de capa de extracción.

- En la figura 5.18 se muestra el resultado de acceso de manera recursiva al contenido de dichos ficheros resultantes:

```

TEXTO ORIGINAL ->
Algebra Booleana El Algebra de Boole permitió establecer la analogía entre los símbolos algebraicos y aquellos que representan sus formas lógicas. Algunos aspectos de interés sobre ello: El álgebra Booleana o de Boole, consiste en un método para resolver problemas de lógica empleando para ello valores binarios 1 y 0 y a tres operadores: AND (y), OR (o) y NOT (no). El álgebra Booleana se emplea en la construcción de computadoras, circuitos eléctricos, etc. Es fundamental destacar que los sistemas modernos trabajan a partir de lógica binaria. En términos generales, el álgebra de Boole se puede emplear a cualquier sistema en que cada variable posea dos estados (1 y 0). Los computadores modernos realizan sus operaciones y almacenan la información empleando valores binarios (1 y 0) y realizan sus operaciones en base al álgebra de Boole. Se emplea en el diseño de circuitos electrónicos. Se emplea ampliamente en la programación (condiciones lógicas). Te invitamos a ver este video, elaborado por RTE-Ireland's National Public Service Media, que presenta la importancia del Álgebra Booleana para la tecnología de hoy en día.

TEXTO ORIGINAL ->
George Boole (1815 - 1864) fue un famoso matemático de origen inglés que publicó un tratado sobre las leyes del pensamiento, el cual sustenta las teorías de la lógica y la probabilidad. El siguiente video elaborado por BBC History School (2015) al celebrar el bicentenario del nacimiento de George Boole te permitirá conocer un poco más sobre su vida. Si quieres conocer los eventos que ocurrieron a lo largo de la vida de George Boole, te invitamos a revisar el siguiente enlace George Boole 200

TEXTO ORIGINAL ->
Para el proceso de simplificación se puede agrupar los 1 en potencias de dos (1, 2, 4 u 8) celdas como se visualiza en la figura. A fin de simplificar la expresión, deberemos seguir estas reglas: Las celdas que contienen 1 se conocen como minterms y las que tienen 0 maxterms. Estas se deberán agrupar de acuerdo a las funciones lógicas con las que trabajemos. No se pueden hacer agrupaciones en diagonal, las celdas que estén en los bordes (por ejemplo primera fila) podrán agruparse con las de la última fila (misma columna). A fin de determinar la ecuación lógica resultante, se deben considerar aquellas variables que no cambian dentro del grupo. Las que si lo hagan se eliminan.

```

Figura 5.18 Extracción recursiva de contenido de ficheros.

- En la figura 5.19 se puede apreciar la identificación de ID's correspondientes a reco-mendaciones dadas por el S.E:

```

Resultado de recomendar desde FILE :
{u'id': 40, u'name': 'u'No se debe utilizar el punto y coma (;)'}
{u'id': 43, u'name': 'u'No se debe utilizar etc\wetera, ni los puntos suspensivos (\u2026). Se puede sustituir por entre otros, y muchos m\wels y otras frases similares.'}
{u'id': 41, u'name': 'u' Se deber\weda evitar el uso de par\wentesis, corchetes y signos ortogr\weficos poco habituales (% , & , /, \u2026, etc.).'}
{u'id': 2, u'name': 'u' Se debe utilizar la may\wfascula inicial al principio de un p\wexrafo o un t\weditulo, despu\wexs de punto o en nombres propios.'}
[02/Sep/2019 07:55:24] "POST /api/preferences HTTP/1.1" 200 1289

```

Figura 5.19 Identificación de ID's de configuraciones resultantes del S.E.

- A continuación se procede a emparejar cada Id de recomendación con cada método programado en el módulo de procesamiento, definiendo un identificador para cada método que servirá para realizar un “matching”, y proceder a ejecutar cada uno de ellos proporcional a las recomendaciones brindadas.

En el desarrollo de este trabajo se plantea el uso de spaCy como una alternativa libre para el procesamiento avanzado del lenguaje natural. “spaCy está diseñado específicamente para uso en producción y lo ayuda a crear aplicaciones que procesan y “entienden” grandes volúmenes de texto. Se puede utilizar para construir sistemas de

extracción de información, comprensión del lenguaje natural, o para preprocesar texto para el aprendizaje profundo” [21].

La implementación de esta biblioteca se la toma en consideración basados en el potencial y las características que posee, entre las cuales se puede identificar [21]:

- Tokenización: permite realizar una segmentación de textos en palabras, signos de puntuación, etc.
 - Análisis de dependencia: genera una asignación de “etiquetas de dependencia sintácticas, describiendo las relaciones entre token individuales, como sujeto y objeto”.
 - Detección de límite de oración (SBD): Encontrar y segmentar oraciones individuales.
 - Reconocimiento de entidad con nombre (NER): Etiquetado de objetos denominados “real-world”, como personas, empresas o ubicaciones.
 - Semejanza: Comparando palabras, espacios de texto y documentos cuán similares son entre sí.
 - Emparejamiento basado en reglas: Encontrar secuencias de tokens basadas en sus textos y anotaciones lingüísticas, similares a las expresiones regulares.
- Y otras características más con función independiente.

Dentro de estas estructuras y sobre las cuales se basa nuestro análisis de procesamiento de lenguaje se encuentran los objetos contenedor Span como una parte de un Doc y Token como una palabra, un símbolo de puntuación o un espacio en blanco. Gracias a estos objetos la capa de procesamiento identifica determinadas anotaciones lingüísticas en los textos extraídos.

A continuación en la Figura 5.20 se puede identificar las estructuras de datos centrales en spaCy, Doc y Vocab. “El objeto Doc posee la secuencia de tokens y todas sus

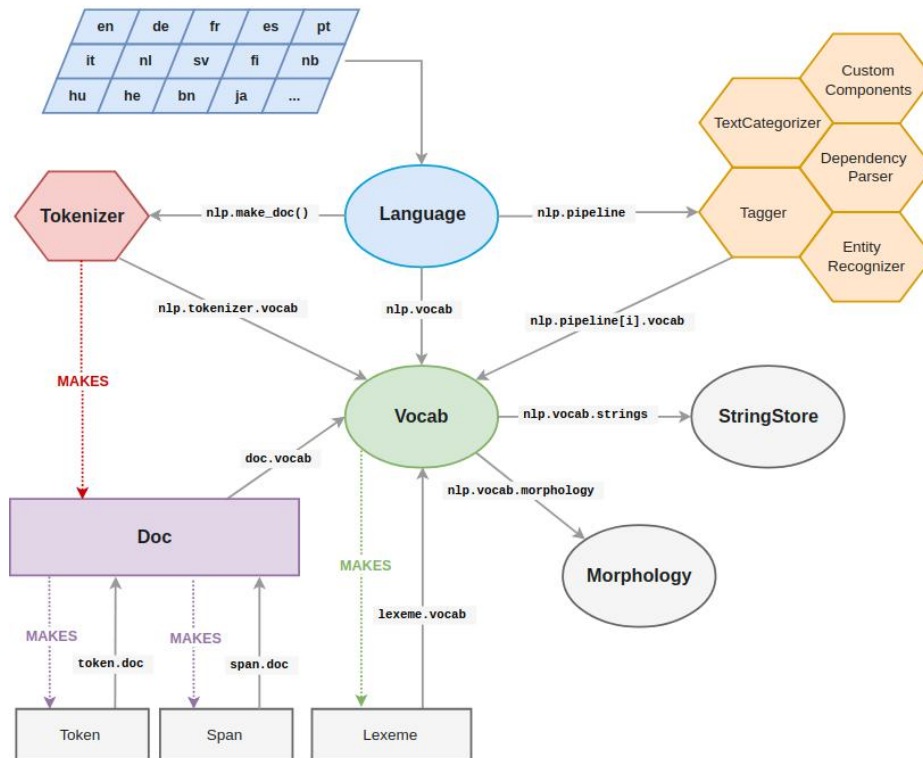


Figura 5.20 Estructuras de datos centrales de spaCy (Cortesía: spaCy, 2019) [21].

anotaciones. El objeto Vocab posee un conjunto de tablas de consulta que hacen que la información común esté disponible en todos los documentos. Al centralizar cadenas, vectores de palabras y atributos léxicos, evitamos almacenar múltiples copias de estos datos. Esto ahorra memoria y asegura que haya una sola fuente de verdad” [21].

5.5.1.3.1. Entrenamiento del modelos de lenguaje spaCy: Basados en la arquitectura de datos centrales (Ver Figura 5.20), spaCy tiene la gran ventaja de poder personalizar y entrenar nuevos modelos estadísticos que potenciar el etiquetador de habla de spaCy, y al tener un modelo entrenado poderlo almacenar para su posterior uso (Ver Figura 5.21). Cabe recalcar que los modelos de lenguaje de spaCy son estadísticos y cada asignación de etiquetas a un texto o un matching entre una palabra con una organización o entidad se la interpreta como una predicción, donde esta se basa en entrenamientos o ejemplos entrenados a partir de textos, etiquetas que se le

indica al modelos que debe predecir, es donde una primera etapa de desarrollo de este trabajo se emplea el entrenamiento de etiquetas. Para lograr dicho cometido se le presenta al modelo un texto sin etiquetar y este realizará la predicción basado en un diccionario de entrenamiento brindándole al modelo una retroalimentación sobre dicha predicción. Para verificar que el resultado de la predicción se acerca a lo que deseamos lograr, spaCy se basa en un "gradiente de error de la función de perdida" que calcula la diferencia entre el ejemplo de entrenamiento y el resultado esperado.

“Cuanto mayor es la diferencia, más significativo es el gradiente y las actualizaciones de nuestro modelo” [21].

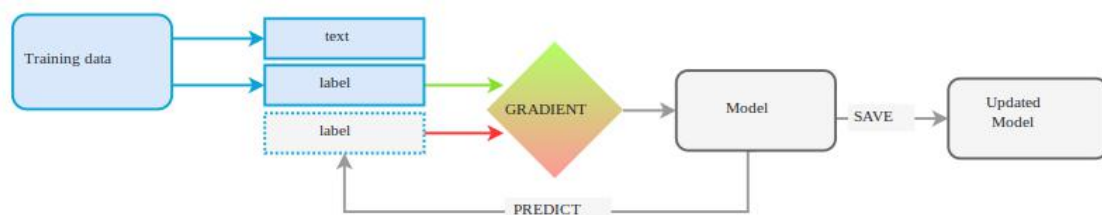


Figura 5.21 Estructura de predicción para modelos de entrenamiento spaCy (Cortesía: spaCy, 2019) [21].

Es importante mencionar que spaCy basa sus fuentes de modelos de lenguaje de Greek Dependency Treebank, Daras GSOC 2018, Universal Dependencies, AnCora y Wikipedia.

Para entrenar nuestro modelo de lenguaje empleamos un “estilo sencillo de formación” que consiste en pasar como parámetro textos primarios y diccionarios de anotación a una tubería `nlp.update`

Se procede con el entrenamiento de nuestro modelo y nos centramos en la etiqueta `entities` evitando importaciones innecesarias que consumen recursos y tiempo, como por ejemplo `heads`, `deps`, `tagsy` y `cats`. Cabe recalcar que esto dependerá mucho del escenario sobre el que se quiera desarrollar y trabajar.

A continuación definimos 4 pasos para lograr dicho entrenamiento:

- Cargar el modelo: en este paso empleamos un modelo de lenguaje existente en este caso Spanish (Vocabulary, syntax, entities, vectors), y procedemos a deshabilitar algunos componentes de tubería para entrenar únicamente el reconocedor de entidad, indicando para ello la sentencia `disable_pipes`.
- Shuffle and loop over: para cada ejemplo definido se realiza un llamado a `nlp.update`, recorriendo cada palabra de este ejemplo realizando en cada una de ellas una predicción, consultando en cada paso las anotaciones para validar que sean correctas y ajustando los pesos en el caso de que sean erróneas con el objetivo de obtener una puntuación más significativa en la próxima predicción.
- Almacenamiento del modelo: una vez que se obtiene una predicción considerable procedemos a almacenar nuestro modelo para cargarlo en un nuevo escenario. Empleado para ello la sentencia `nlp.to_disk`
- Prueba del modelos: al almacenar el modelo entrenado procedemos a cargarlo en nuestro escenario y verificamos que se reconozcan correctamente las etiquetas entrenadas.

En el código 5.14 podemos identificar las sentencias empleadas para lograr un entrenamiento en el reconocimiento de entidades.

```

1 TRAIN_DATA = [("La ONG", {"entities": [(3, 6, "ORG")]}), ("la ONG Paz
2 Ahora", {"entities": [(3, 16, "ORG"),]}), ]
3 plac.annotations(model=("Nombre del modelo. Default para modelo en
4 blanco'es' model.", "option", "m", "es"),
output_dir=("Directorio de salida opcional", "option", "o", "/usr/
local/lib/python2.7/dist-packages/spacy/data/es_core_news_md/
es_core_news_md-2.0.0"),
n_iter=("Numero de interacciones de entrenamiento", "option", "n",
10),)

```

```
5 # ***** M todo para el entrenamiento de entidades para el
   modelo en espa ol *****
6 def train_entities(self, model=None, output_dir=None, n_iter=100):
7     global nlp2
8     output_dir="/usr/local/lib/python2.7/dist-packages/spacy/data/
   es_core_news_md/es_core_news_md-2.0.0"
9     if model is not None:
10        nlp = spacy.load(model) #se realiza la carga del modelo de spaCy
   existente
11    else:
12        nlp = spacy.blank("es") #si no se identifica modelo se crea una
   clase de lenguaje en blanco
13        print("Creando el modelo 'es' ")
14        #Proceso de creaci n de los componentes de canalizaci n
   incoportados y se los agrega al pipeline (tuberia)
15        if "ner" not in nlp.pipe_names:
16            ner = nlp.create_pipe("ner")
17            nlp.add_pipe(ner, last=True)
18            #De lo contrario, se agregan etiquetas.
19        else:
20            ner = nlp.get_pipe("ner")
21        #agregaci n de etiquetas
22        for _, annotations in TRAIN_DATA:
23            for ent in annotations.get("entities"):
24                ner.add_label(ent[2])
25        # Proceso para obtener nombres de otras tuber as para
   deshabilitarlas durante el entrenamiento
26        otras_pipes = [pipe for pipe in nlp.pipe_names if pipe != "ner"]
27        with nlp.disable_pipes(*otras_pipes): # solo entrenamiento de NER
28            # entrenando un nuevo modelo
29            if model is None:
30                nlp.begin_training()
```

```
31     for itn in range(n_iter):
32         random.shuffle(TRAIN_DATA)
33         losses = {}
34         # agruparlos ejemplos usando minibatch de spaCy
35         batches = minibatch(TRAIN_DATA, size = compounding(4.0, 32.0,
36     1.001))
37         # Cada vez que se llama al generador, se genera un valor
38     multiplicando el valor anterior por la tasa compuesta.
39         for batch in batches:
40             texts, annotations = zip(*batch)
41             nlp.update(
42                 texts, # lote de textos annotations, # lote de
43     anotaciones
44                 drop=0.5, # dropout-hace que seamos difciles
45     memorizar los datos
46                 losses=losses,)
47             print("Perdidas", losses)
48     # Test del modelo de entrenamiento
49     for text, _ in TRAIN_DATA:
50         doc = nlp(text)
51         print("Entidades", [(ent.text, ent.label_) for ent in doc.ents])
52         print("Tokens de Entidades", [(t.text, t.ent_type_, t.ent_iob)
53     for t in doc])
54     # guarda del modelo en el directorio de salida de spaCy
55     if output_dir is not None:
56         output_dir = Path(output_dir)
57         if not output_dir.exists():
58             output_dir.mkdir()
59         nlp.to_disk(output_dir)
60         print("Modelo guardado en", output_dir)
61     # prueba del modelo guardado
62     print("Carga del modelo desde", output_dir)
```

```

58     nlp2 = spacy.load(output_dir)
59     print("VALOR DE NLPS ", nlp2)
60     for text, _ in TRAIN_DATA:
61         doc = nlp2(text)
62         print("Entidad encontrada : -> ", [(e
        nt.text, ent.label_) for
ent in doc.ents])
63         print("Tokens entidades : -> ", [(t.tex
t, t.ent_type_, t.
ent_iob) for t in doc])
64     filepath = '/home/roly/Documentos/tesis/corpus.txt'
65     print("LLAMADA A SEGUNDA FUNCION")
66     plac.call(entities_analyzer(self, filepath))
        self.entities_analyzer(filepath)
68

```

Código 5.14 Ejemplo de entrenamiento de entidades sobre modelo de lenguaje español de spaCy.

Este entrenamiento se lo realiza cuando requerimos que las etiquetas del modelo de lenguaje se apeguen a nuestra necesidad, por ejemplo entidades que no aparecen en un modelo de lenguaje importado, que es el caso de realizar el entrenamiento en nuestro desarrollo, enriqueciendo así el procesamiento de lenguaje natural y ampliando el esquema propuesto. Se recalca que spaCy tiene una extensa documentación y se ha dado una breve instrucción de sus características, importancia y potencial por el cual fue empleado para el desarrollo de este trabajo.

Para el uso de spaCy como biblioteca de procesamiento de lenguaje se realiza un llamado de la biblioteca así como el modelo de lenguaje de spaCy a emplear, en este caso el español, como se lo puede apreciar en el código 5.15:

```

import spacy

```

import textacy

```

3 import re
4 import unicodedata
5 import es_core_news_sm
6 from spacy import displacy
7 from spacy.util import minibatch, compounding

```

Código 5.15 Uso de spaCy dentro de la capa de procesamiento.

Una vez que se tiene la biblioteca y el modelo definido se pone en marcha el uso de sus características dependiendo del contexto requerido, se puede apreciar en el código 5.16 como se hace uso de estas técnicas para la identificación de signos poco habituales.

```

1 def orthographic_signs_analyzer(self, text):
2     doc = nlp(text)
3     for token in doc:
4         print("{} | -> {}".format(token, token.tag_, token.
5 pos_, token.dep_))
6     # DETERMINACION DE SIGNOS ORTOGRAFICOS POCO HABITUALES
7     print("TEXTO ORIGINAL ->", doc.text)
8     for token in doc:
9         if token.tag_ == 'PUNCT__PunctType=Semi':
10            text = text.replace(";", ".(*)")
11        if token.tag_ == 'SYM__NumForm=Digit|NumType=Frac':
12            text = text.replace("%", "porciento(*)")
13        if token.tag_ == 'NOUN__':
14            text = text.replace("$", "dolar/es")
15        if token.tag_ == 'PUNCT__PunctType=Comm':
16            text = text.replace("etc", "entre otros(*)")
17            text = text.replace("...", ", y muchos m s.(*)")
18        if token.tag_ == 'PROPN__':
19            text = text.replace("&", "y(*)")
20        if token.tag_ == 'PUNCT__PunctSide=Ini|PunctType=Brck':

```

```
20 text = text.replace("(", ",");
```

Código 5.16 Ejemplo de método para analizar signos poco habituales con el uso de spaCy.

Para cada método de análisis de textos se hace uso de esta biblioteca de procesamiento de lenguaje natural en combinación con expresiones regulares, como por ejemplo en el código 5.17.

```
1 mente reg=re.match(r"(.+?)\mente$|(.+?)\mente.|(.+?)\  
mente,$",tok.text);
```

Código 5.17 Identificación de adverbios terminados en -mente

Esta capa básicamente se constituye del uso de spaCy para identificación de tokens y expresiones regulares para determinados procesamientos, basado en la configuración de la normativa usada. Al finalizar este procesamiento se obtiene como resultado un texto analizado y limpio a más de ello adaptado a las pautas y recomendaciones de lectura fácil para una persona con un perfil de discapacidad determinado. A continuación se evidencian esta identificación de tokens al aplicar este análisis (Ver Figura 5.22) :

El resultado de este procesamiento es plasmado en la interface con el docente para su análisis y edición de ser necesario.

5.5.1.4. Análisis de contrastes:

Para cubrir con las pautas de la normativa UNE:170002 que trata sobre el manejo de contrastes en el uso de fondos y textos, empleamos un análisis automatizado, el mismo que permite extraer y analizar los colores usados en fondos del O.A y un análisis general representado en:

- Cuantificación de píxeles x colores encontrados en el O.A.


```

lenix@lenix-PC: ~/Documents/tesistemas
(u'Texto citado encontrados', "(u'En la Figura podemos observar su tabla de verdad, el símbolo que la representa y su ecuación correspondiente', u'')")
Segun la normativa UNE 153101:2018 EX las citas textuales deben ir acompañadas de una explicación.
(u'Texto citado español encontrado/s', '[]')
(u'Los rufameros con muchos dígitos son difíciles de leer y escribirlos con letra puede facilitar su lectura.', u'8.000')
(u'Adverbios terminados en mente encontrados:', u'correspondientemente(*)')
(u'Los rufameros con muchos dígitos son difíciles de leer y escribirlos con letra puede facilitar su lectura.', u'100,124.55556')
asunto
(u'Adverbios terminados en mente encontrados:', u'completamente(*)')
(u'Fecha encontrada:', u'2020/04/12', u', se debería evitar escribir la fecha con guiones o barras')
(u'Formatos de hora encontrados', u'23:00')
(u'Se han encontrado MAYUSCULAS', [u'NOT', u'AND', u'OR', u'PDF', u'NOR', u'NAND', u'OR', u'OR', u'AND', u'AND', u'NOT', u'NOT', u'AND', u'OR', u'NOT', u'RTxc9', u'UCC', u'OR', u'AND', u'NOT', u'NAND', u'NOR', u'XOR', u'OR', u'XNOR', u'NOR', u'XOR', u'NOR', u'NOR', u'OR', u'NOT', u'NAND', u'NAND', u'NOR', u'XNOR', u'XOR', u'NOT'])
/txtp/ Compuerta XOR Compuerta ,XOR La operación que se lleva a cabo con esta compuerta devolverá en la salida un valor lógico de 1 u ocho mil punto cero siempre y cuando las dos entradas sean distintas. Esta compuerta recibe dos entradas 100 por ciento. En la Figura podemos observar su tabla de verdad, el símbolo que la representa y su ecuación correspondiente. Compuerta NOR Compuerta ,NOR Representa la combinación de dos compuertas: una OR seguida de una NOT. Es importante observar que esta compuerta recibe dos entradas. En la figura podemos $ observar su tabla de verdad, el símbolo que la representa y su ecuación correspondiente. Compuerta NAND Compuerta ,NAND Representa la combinación de dos compuertas. Es importante observar que esta compuerta recibe dos entradas, y muchos más. En la figura 1/2 podemos observar su tabla de verdad, el símbolo que la representa y su ecuación correspondiente. Compuerta XNOR Compuerta ,XNOR Representa la combinación de dos compuertas: una XOR seguida de una NOT. Es importante observar que esta compuerta recibe dos objetos 12 de abril del 2020 (*) 11:00 de la noche En la figura diez mil doce millones cuatrocientos cincuenta y cinco mil quinientos cincuenta y seis punto cero podemos observar su tabla de verdad, el símbolo que la <> y, su ecuación correspondiente(*). Esta compuerta se puede representar mediante la equivalencia tema que se presenta en la siguiente ecuación: completa. temas
    
```

Figura 5.25 Resultado de aplicación de recomendaciones dadas por el S.E sobre texto del O.A.

Resultado del Sistema Experto
 Basado en sus necesidades el sistema experto le proporciona las siguientes configuraciones que le ayudaran a adaptar su contenido

(u'Id: 40, u'pauta: u'No se debe utilizar el punto y coma', u'Name: u'No se debe utilizar el punto y coma (,)', u'Id: 43, u'pauta: u'No se debe utilizar etcetera', u'Name: u'No se debe utilizar etcetera, ni los puntos suspensivos (u2026). Se puede sustituir por entre otros, y muchos mve1s y otras frases similares.', u'Id: 41, u'pauta: u'Se debería evitar el uso de parentesis', u'Name: u' Se debería evitar el uso de paréntesis, corchetes y signos ortográficos poco habituales (% & /u2026, etc.)) (u'Id: 2, u'pauta: u'1 Se debe utilizar la mayúscula inicial al principio de un párrafo o un título', u'Name: u' Se debe utilizar la mayúscula inicial al principio de un párrafo o un título, después de punto o en nombres propios.)

Configuraciones aplicadas
 Según la normativa UNE 153101 se procede a indicar cuales textos están escritas en Lectura Fácil mediante(*)

Compuertas_Derivadas (u'Texto citado encontrados', "(u'En la Figura podemos observar su tabla de verdad, el símbolo que la representa y su ecuación correspondiente', u'') Segun la normativa UNE 153101:2018 EX las citas textuales deben ir acompañadas de una explicación. (u'Texto citado español encontrado/s', []) (u'Los rufameros con muchos dígitos son difíciles de leer y escribirlos con letra puede facilitar su lectura.', u'8.000') (u'Adverbios terminados en mente encontrados:', u'correspondientemente(*)') (u'Los rufameros con muchos dígitos son difíciles de leer y escribirlos con letra puede facilitar su lectura.', u'100,124.55556') asunto (u'Adverbios terminados en mente encontrados:', u'completamente(*)') (u'Fecha encontrada:', u'2020/04/12', u', se debería evitar escribir la fecha con guiones o barras') (u'Formatos de hora encontrados', u'23:00') (u'Se han encontrado MAYUSCULAS', [u'NOT', u'AND', u'OR', u'PDF', u'NOR', u'NAND', u'OR', u'AND', u'AND', u'NOT', u'NOT', u'AND', u'OR', u'NOT', u'RTxc9', u'UCC', u'OR', u'AND', u'NOT', u'NAND', u'NOR', u'XOR', u'OR', u'XNOR', u'NOR', u'XOR', u'NOR', u'NOR', u'OR', u'NOT', u'NAND', u'NAND', u'NOR', u'XNOR', u'XOR', u'NOT'])

Textos procesados

Resultado de procesamiento
 — Pautas para configuraciones aplicadas

← Anterior Página Siguiete Página →

Compuertas Derivadas

— Texto Original	— Texto Procesado
Compuerta XOR Compuerta (XOR) La operación que se lleva a cabo con esta compuerta devolverá en la salida un valor lógico de 1 u 8.000 siempre y cuando las dos entradas sean distintas. Esta compuerta recibe dos entradas 100%. "En la Figura podemos observar su tabla de verdad, el símbolo que la representa y su ecuación correspondiente". Compuerta NOR Compuerta (NOR) Representa la combinación de dos compuertas: una OR seguida de una NOT. Es importante observar que esta compuerta recibe dos entradas. En la figura podemos \$ observar su tabla de verdad, el símbolo que la representa y su ecuación correspondiente,etc. Compuerta NAND Compuerta (NAND) Representa la combinación de dos compuertas; una "AND" seguida de una NOT. Es importante	Compuerta XOR Compuerta XOR La operación que se lleva a cabo con esta compuerta devolverá en la salida un valor lógico de 1 u ocho mil punto cero siempre y cuando las dos entradas sean distintas. Esta compuerta recibe dos entradas 100 por ciento . En la Figura podemos observar su tabla de verdad, el símbolo que la representa y su ecuación correspondiente. Compuerta NOR Compuerta, NOR Representa la combinación de dos compuertas: una OR seguida de una NOT . Es importante observar que esta compuerta recibe dos entradas. En la figura podemos observar su tabla de verdad, el símbolo que la representa y su ecuación correspondiente,entre otros. Compuerta NAND Compuerta, NAND Representa la combinación de dos compuertas. Una AND

Figura 5.26 Resultado de proceso de adaptación previo a ensamblaje.

- Porcentajes de colores usado en el O.A.
- Una representación en paleta de colores para indicar al docente qué combinaciones de colores se apegan a las buenas practicas y cuales no.

Para el desarrollo de este análisis se trabaja en función de las similitudes visuales usando la formula CIE76. Cabe indicar que dicha etapa se la puede perfeccionar con el empleo de mejores algoritmos y realizar un análisis más profundo, pero para dar un breve acercamiento al objetivo de las pautas indicadas es válido. El funcionamiento de esta formula consiste en encontrar una matriz de color que agrupa los colores percibidos en el O.A y proceder a medir la distancia entre los puntos definidos en un espacio 3D CIELAB, empleado para dicha diferencia la fórmula Delta E similar a la distancia euclidiana [26]:

$$\Delta E_{a^* b^*} = \sqrt{(L_2^* - L_1^*)^2 + (a_2^* - a_1^*)^2 + (b_2^* - b_1^*)^2} \quad (5.1)$$

Al interpretarla en una escala Delta E variará de 0 a 100.

Delta E	Percepción
<= 1.0	No perceptible por los ojos humanos.
1 - 2	Perceptible a través de la observación cercana.
2 - 10	Perceptible de un vistazo.
11 - 49	Los colores son más parecidos que los opuestos.
100	Los colores son exactamente opuestos

Cuadro 5.9 Escala Delta E [26]

Para lograr la extracción de esta escala nos basamos en una biblioteca empleada en OpenCV, esta biblioteca proporciona la facilidad de identificar un matriz de color para su posterior análisis y cuantificación de porcentajes de colores en cada componente html del O.A, para ello nos basamos en las imágenes obtenidas con el navegador sin cabecera y técnicas empleadas en PhantomJs, generando como resultado una captura de pantalla de cada archivo html que constituye el O.A. (Ver Figura 5.27)

Misma que se le presenta al usuario brindándole una guía del proceso secuencial que se realiza (Ver Figura 5.28, 5.29).



Figura 5.27 Screen de recursos del O.A a partir de navegador sin cabecera para análisis de contrastes.

Cabe indicar que este análisis se lo realiza en conjunto con las recomendaciones dadas por el Sistema Experto.

En paralelo con la presentación los textos procesados, brindamos al Docente la capacidad de identificar si la combinación de colores empleados con los fondos del O.A son adecuados o no:



Figura 5.28 Resultado de identificación de contrastes en componente html con su correspondiente escala de colores

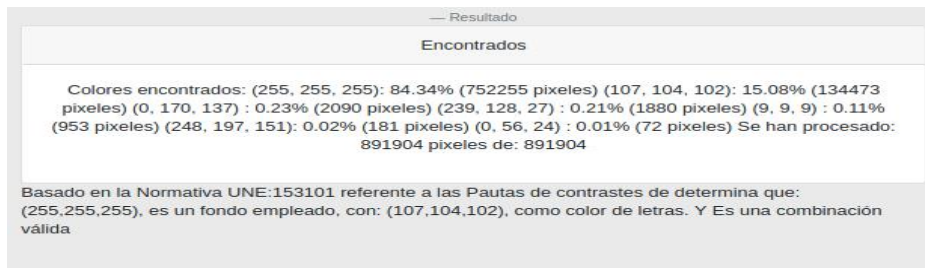


Figura 5.29 Escala de contraste identificando color, porcentaje de color y cantidad de pixeles por color

5.5.1.5. Capa de ensamblaje

Con el desarrollo de esta capa se presenta al usuario final un recurso personalizado y adaptado para personas con discapacidad basado en normativas de accesibilidad, asociando los resultados de todos los procesos definidos en las capas anteriores en un solo material que puede ser impartido en cualquier sistema de gestión de aprendizaje. Para lograr este cometido se tiene en cuenta el formato en el que va a ser presentado dicho resultado, siendo este un ePub (Electronic Publication) que básicamente es un formato de archivo e-book que es reconocido en cualquier lector electrónico y comúnmente usado en la mayoría de LMS.

Entre las características más relevantes que se encuentran en un ePub podemos mencionar su formato de texto redimensionable, esto quiere decir que se puede adaptar y brindar la facilidad de seleccionar entre distintos tamaños tanto de letra como de pantalla para ser proyectado.

Este material puede ser empleado con herramientas de terceros que funcionan de TTS (Text to Speech), brindando facilidades en la interpretación de contenidos para personas con discapacidad visual, así también como soportes en contrastes para personas con problemas visuales y cognitivos.

Al definir el formato en el que se presentara al usuario el ensamblaje de cada proceso resultante de aplicar las técnicas descritas en cada capa, se procede a estructurar dicho material.

- Al aplicar las modificaciones realizadas por el docente sobre el material adaptado se procede a almacenar dichos cambios en ficheros texto constituidos por un identificador (title) y su contenido (text), que al instante que el docente aplica los cambios estos ficheros son sometidos a un proceso de ensamblaje de contenido constituido por la siguiente lógica de desarrollo:
 - Identificación ficheros: mediante un controlador desarrollado en python se identifican los ficheros resultantes de la verificación y validación del texto adaptado a las configuraciones de lectura facil, en este proceso se genera una taxonomía de identificadores con su correspondiente contenido validado a raíz de estos ficheros resultantes.
 - Generación de spine: una vez que se obtiene esta taxonomía se procede a generar el spine de nuestro ePub, este spine no es nada más que la estructura que tendrá el nuevo material, en el cual se registran:
 - Metadatos: mediante el uso de ebooklib se agregan los metadatos a nuestro ePub (Ver código 5.18), para ello procedemos a importar dicha biblioteca y definimos los parámetros necesarios para su empleo:

```
1      from ebooklib import epub
2      # Metodo para la generacion de epub final
3      def generate_epub(self, title, txt):
4          book = epub.EpubBook()
5          book.set_identifier('tester03019')
6
7          book.set_title('Epub SIO - Lectura Facil')
8          book.set_language('es')
9          book.add_author("Christian Oyola")
```

Código 5.18 Definición de metadatos en ePub

- Intro: en este apartado se da una breve descripción del material desarrollado, así como el objetivo del mismo:

```

1      ci = epub.EpubHtml(title="Introducción", file_name="
      intro.xhtml", lang="es")
2      ci.content = u'<html><head></head><body><h1>
      Introducción                               n</h1><p>' + {      Description
      ePub                                       }+ '</p
      ></body></html>'

```

Código 5.19 Definición de objetivos y descripción de ePub

- Estilo: para que el resultado final tenga una presentación válida para los estudiantes, se definen ciertas propiedades de estilo que harán de nuestro ePub un material llamativo y funcional (Ver código 5.20).

```

1 estilo= ''' body { font-family: Cambria, Liberation
      Serif,
      Bitstream Vera Serif, Georgia, Times, Times
      New Roman,
2      serif;}
      h2{ text-align:left; text-transform: upper
3      case; font-
4      weight : 200;}
      ol{ list-style-type: none;}
5      ol > li:first-child { margin-top: 0.3em;}
      nav[epub|type ~='toc'] > ol > li > ol { list-style-type
6      :square;}
      nav[epub|type ~='toc'] > ol > li > ol > li {
7      margin-top: 0.3em;} '''
8      nav_css = epub.EpubItem(uid="style_nav", file_name=
      "style/
      nav.css", media_type="text/css", content=estilo)
9

```

```
book.add_item  
m(nav_css)
```

Código 5.20 Definición de estilos de ePub

- Generación de contenido: al definir nuestro spine procedemos a estructurar el contenido de nuestro ePub, en el cual se evidencia todo el proceso de adaptación y validación por parte del sistema y docentes respectivamente, para ello se emplea un proceso cíclico basado en la taxonomía definida, y mediante las propiedades de ebooklib se procede a generar este contenido (Ver código 5.21).

```

1 #definicion de capitulo de epub
2 c_i = epub.EpubHtml(title=title, file_name=title+".xhtml", lang
   = "es")
3 c_i.content = u'<html><head></head><body><h1> +title+ </h1
   ><p>'+ txt.decode('UTF-8') + ' </p></body></html>'
4 #agregar los capitulos en el libro
5 book.add_item(c_i)
6 #creacion de spine o tabla de contenidos, agrega seccion links
   de los capitulos
7 book.toc = (epub.Link(title+'.xhtml', title, title),(epub.
   Section('CONTENIDOS'),(c_i)))
8 book.add_item(epub.EpubNcx())
9 book.add_item(epub.EpubNav())

```

Código 5.21 Definición de estructura de contenidos del ePub

- Espacio de almacenamiento : al definir nuestra estructura resultante procedemos a determinar la ruta de almacenamiento que tendrá el ePub, para ello se le brinda al docente la opción de asignar dicha ruta y este se almacenara como SIO_LecturaFail.epub (Ver código 5.22):

```

1 script_wd = os.path.dirname(os.path.abspath(__file__))
2 root = Tkinter.Tk()
3 root.withdraw()

```

```

4 currdir = os.getcwd()
5 tempdir = tkFileDialog.askdirectory(parent = root,
   initialdir = currdir, title = 'Por favor seleccione directorio')
6 if len(tempdir) > 0:
7     epub.write_epub(str(tempdir) + '/SIO_LecturaFacil.epub', book, {})
8     print("----->>>>> EPUB DESDE SYSMOA CREADO
9     CON EXITO !!! <<<<<<----- ruta %s" % tempdir)

```

Código 5.22 Definición de almacenamiento de ePub resultante

Mediante el proceso planteado se genera un ePub adaptado y personalizado a lectura fácil enfocado a personas con determinado perfil de discapacidad (Ver Figura 5.30, 5.31).

Dicho proceso es generado a partir del análisis y validación por parte del docente, basados en métricas definidas por expertos.

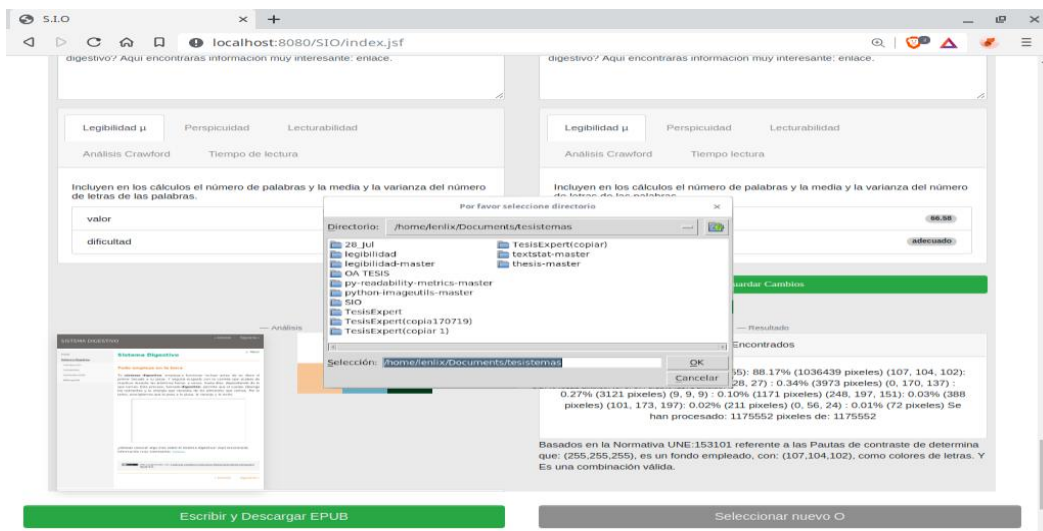


Figura 5.30 Escribir y descargar EPUB posterior validación de textos analizados por Docente

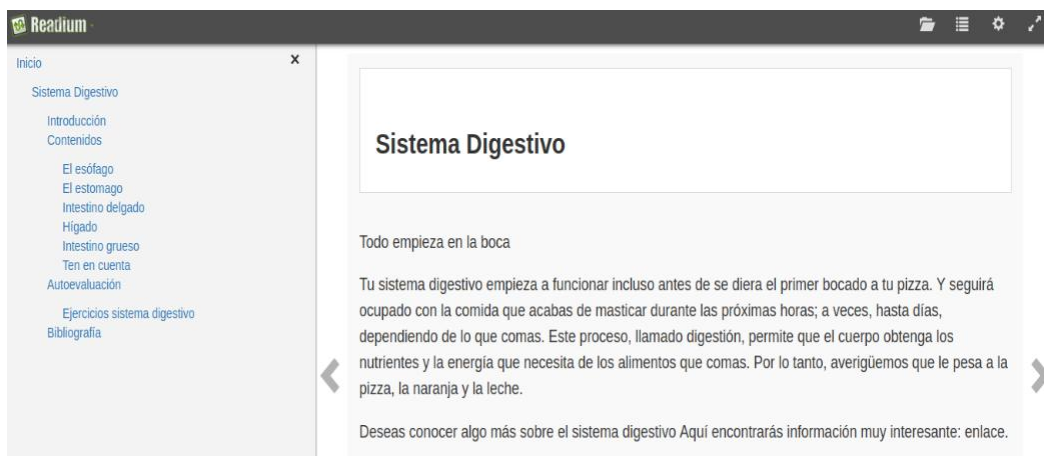


Figura 5.31 Resultado Final de procesamiento del sistema inteligente.

Capítulo 6

Experimentación y resultados

6.1. Plan de experimentación

Con el objetivo de analizar la propuesta se ha definido un proceso de experimentación en el que intervienen algoritmos para medir la legibilidad lingüística. En la primera fase se define 4 algoritmos de evaluación adaptados al lenguaje español y validados mediante investigaciones científicas para determinar la legibilidad del texto, entre estos algoritmos se definen seguidamente.

- Legibilidad μ : por medio de la cual nos permite calcular la facilidad lectora de los textos. Incluyendo un cálculo de número de palabras, la media y la varianza del número de letras encontradas en cada palabra del texto a analizar. Para el cálculo se plantea la siguiente fórmula:

$$\mu = \frac{n}{90} \frac{\bar{x}}{n-1} \sigma^2 * 100 \quad (6.1)$$

Donde: μ es el índice de legibilidad; n , es el número de palabras del texto; \bar{x} , la media del número de letras por palabra (promedio) y σ^2 , su varianza.

Se obtiene una solución similar si se procede a calcular el Coeficiente de variación de Pearson. [1]

Representando esta escala en:

Nivel	Facilidad de lectura	Facilidad de lectura
91 - 100		Muy Fácil
81 - 90		Fácil
71 - 80		Un poco fácil
61 - 70		Adecuado
51 - 60		Un poco difícil
31 - 50		Difícil
0-30		Muy difícil

Cuadro 6.1 Grados de escala μ [1]

- Nivel de perspicuidad - Szigriszt Pazos: es una fórmula diferente capaz de medir la legibilidad pero con la diferencia de que esta es una adaptación a la ecuación de Flesh propuesta por EE.UU. Dicha adaptación se la representa por:

$$P = 206,835 - \frac{63,3S}{P} - \frac{P}{F} \quad (6.2)$$

Donde: P es la perspicuidad; S , es el total de sílabas; P , es la cantidad de palabras y F , es el número de frases [18].

La particularidad de este análisis es que mediante el se puede determinar el estilo inteligible (que puede ser entendido) con el que fue escrito el texto.

P	Estilo	Tipo de publicación	Estudios
0 a 15	muy difícil	científica, filosófica	titulados universitarios
16 a 35	árido	pedagógica, técnica	selectividad y estudios universitarios
36 a 50	bastante difícil	literatura y divulgación	cursos secundarios
51 a 65	normal	Los media	popular
66 a 75	bastante fácil	novela, revista femenina	12 años
76 a 85	fácil	para quioscos	11 años
86 a 100	muy fácil	cómics, tebeos y viñetas	6 a 10 años

Cuadro 6.2 Nivel de perspicuidad (P) [18]

- Escala de lecturabilidad - Fernández Huerta: mediante esta métrica se puede determinar que tan fácil o difícil es un texto de entender. La diferencia es que este análisis omite la presentación tipográfica que es un factor que también influye en la facilidad de lectura. Esta métrica fue adaptada al lenguaje español definida por:

$$L = 206,84 - 0,60P - 1,02F \quad (6.3)$$

Donde: L es la lecturabilidad; P, es el promedio de sílabas por palabra; F, la media de palabras por frase [6].

Representado por:

L	Nivel	Grado escolar
90-100	muy fácil	4º grado
80-90	fácil	5º grado
70-80	algo fácil	6º grado
60-70	normal (para adulto)	7º u 8º grado
50-60	algo difícil	preuniversitario
30-50	difícil	cursos selectivos

Cuadro 6.3 Escala de lecturabilidad (L) [6]

- Comprensibilidad - Crawford : Este algoritmos nos permite proporcionar al docente una referencia de los años de escolaridad se necesitan para comprender determinado

texto generado, con la particularidad de que este valor sirve únicamente para niños de primaria o escuela. Representado por la siguiente ecuación:

$$A = -0,205OP + 0,049SP - 3,407 \quad (6.4)$$

Donde: A es el número de años de escolarización; OP, es el número de oraciones por cien palabras; SP, es el número de sílabas por cien palabras. [4].

Para el empleo de estos algoritmos se plantean diferentes escenarios con los cuales se desea medir cuán óptimo es el sistema para adaptar el contenido de un O.A. para una discapacidad determinada.

Con este objetivo se procede a realizar el análisis con 30 pruebas ligadas a 2 perfiles de discapacidad diferentes representados en el cuadro 6.4:

Discapacidad: Cognitiva			Discapacidad: Auditiva		
In	SI	No	In	SI	No
I1	x		I1	x	
I2	x		I2		x
I3		x	I3		x
I4	x		I4	x	
I5		x	I5	x	
I6		x	I6		x
I7	x		I7	x	
I8		x	I8	x	
I9	x		I9	x	
I10	x		I10	x	
I11	x		I11	x	
I12	x		I12	x	
I13	x		I13		x

Cuadro 6.4 Experimentación con perfiles de discapacidad.

Además se define el formulario de necesidades presentado para cada perfil en el cuadro 6.5.

Indicador	Requerimientos
I1	¿Desea que en su texto se analicen signos poco habituales como %, &, etcétera?
I2	¿Su texto contiene o referencia información de organismos internacionales? Ejm. UNESCO, OMS, ONU, ?
I3	¿Desea que sean procesadas las listas para su mejor comprensión? Ejm (- Elemento 1, - 2 Elemento, -3 Elemento)
I4	¿Desea que en su texto se agregue un glosario o glosa?
I5	¿Desea que en su texto se procese los adverbios y superlativos ?
I6	¿Desea que en su texto se procese abreviaturas y acrónimos ?
I7	¿Su texto contiene palabras de contenido indeterminado como: cosa, algo o asunto?
I8	¿Desea que en su texto se procese las fechas y formatos de horas ?
I9	¿Desea que en su texto se procesen los dígitos numéricos ?
I10	¿Su texto tiene conectores complejos como por lo tanto, no obstante, por consiguiente o sin embargo?
I11	¿Desea que en su texto se procese interlineado, longitud de texto y alineado de texto?
I12	¿Desea procesar los contrastes de color de su texto y fondo?
I13	¿Desea que en su texto se procese el uso de estilos, como cursiva, subrayado?

Cuadro 6.5 Formulario de necesidades presentada al Docente responsable de adaptar el O.A.

- Caso N°1 - O.A. Sistema Digestivo: Se procesa el O.A. Sistema Digestivo identificando 11 componentes Html, siendo estos:

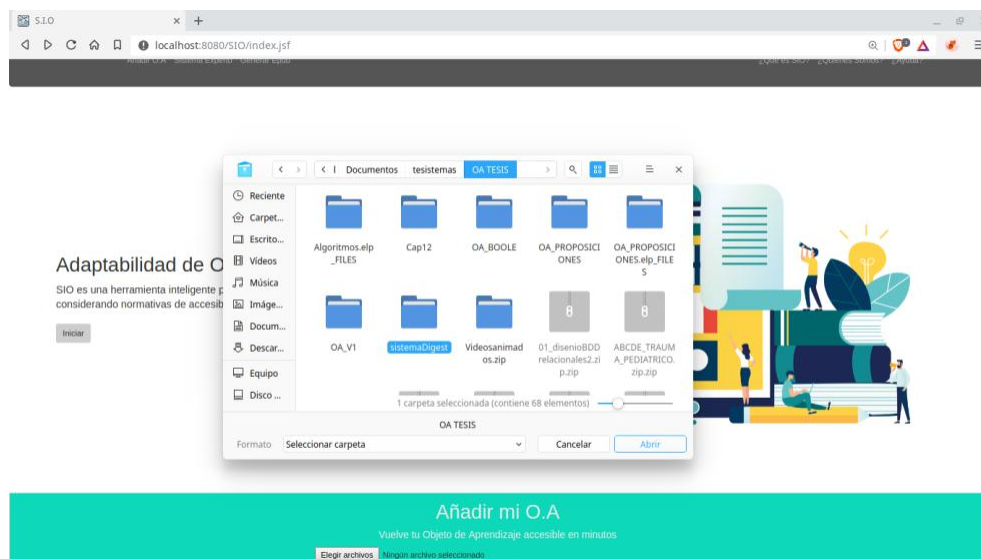


Figura 6.1 Interfaz para carga de O.A. por Docente

Mediante el primer experimento y empleando el sistema experto para la determinación de las mejores configuraciones para estas necesidades se obtiene el resultado presentado en la Figura 6.2:

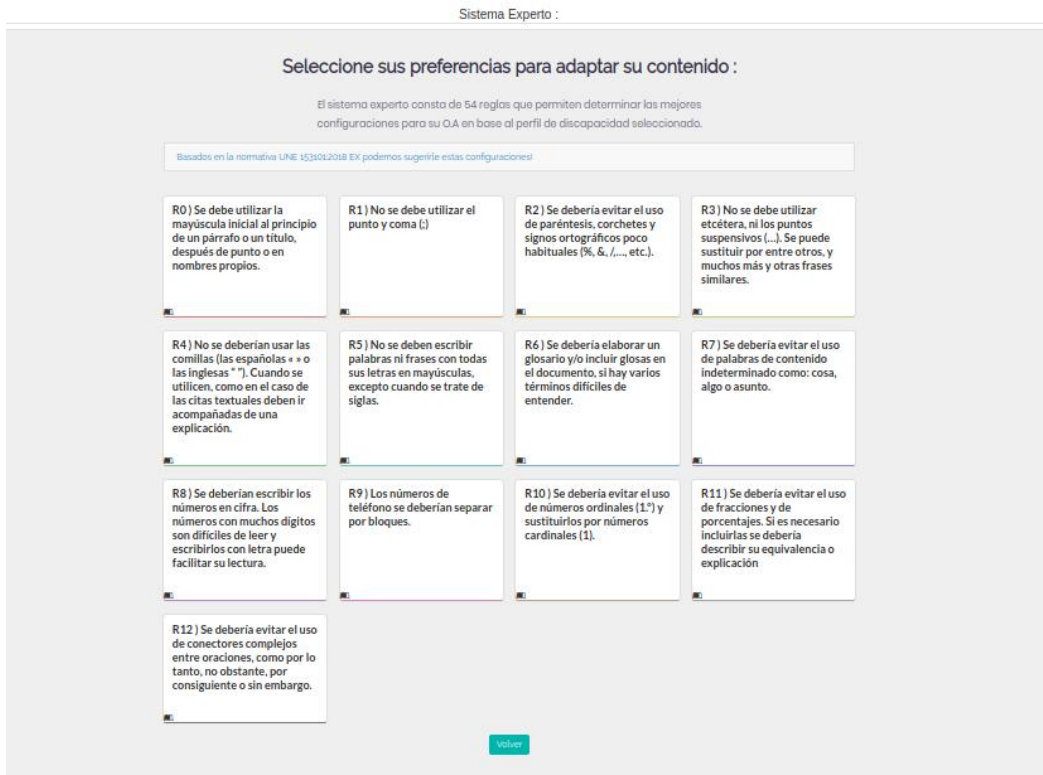


Figura 6.2 Identificación de componentes Html de O.A.

Resultado del análisis del S.E a procesar sobre los textos extraídos de los elementos Html del O.A.

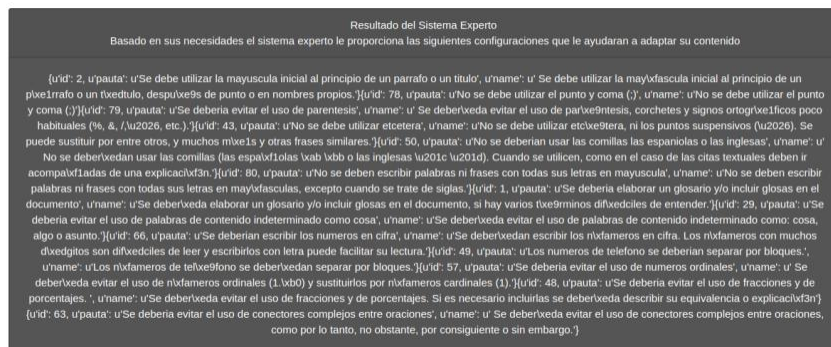


Figura 6.3 Resultado del S.E para experimentación N°1 para discapacidad Cognitiva

Al emplear los algoritmos propuestos sobre los textos originales en los 11 componentes Html de O.A. sistema digestivo, se obtienen los siguientes índices para Legibilidad, Perspicuidad, Lecturabilidad, Crawford.

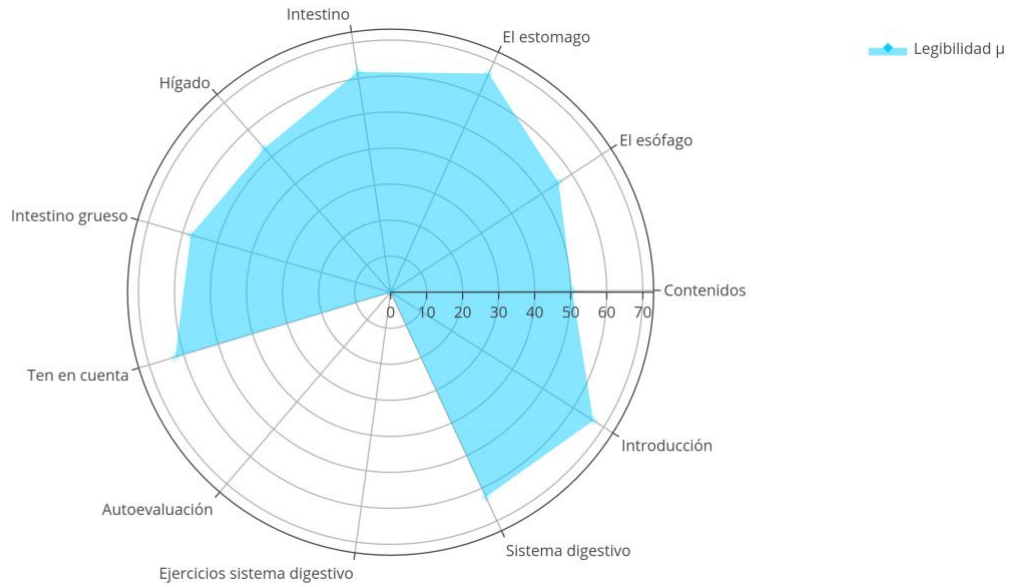


Figura 6.4 Análisis de Legibilidad en textos originales extraídos del O.A.

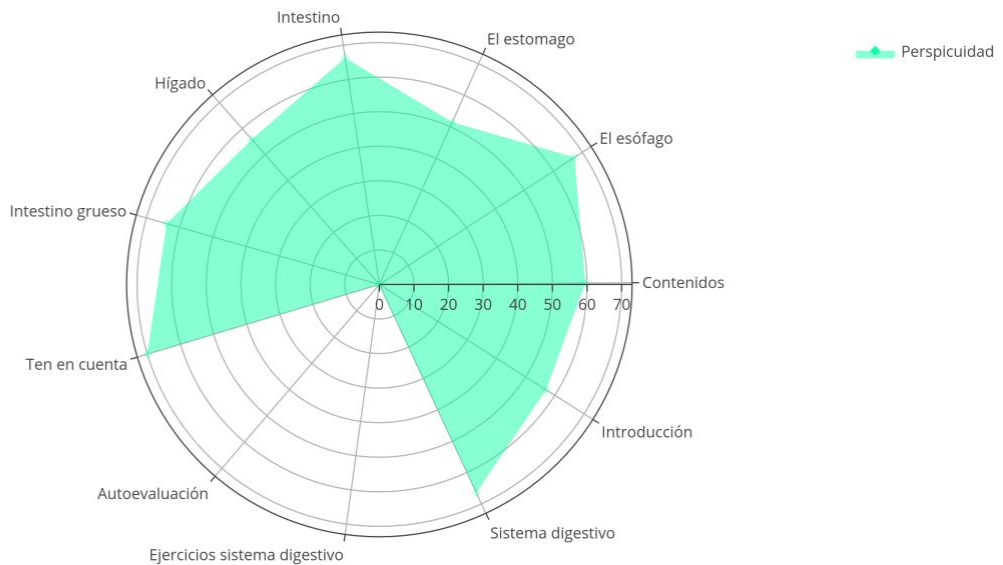


Figura 6.5 Análisis de Perspicuidad en textos originales extraídos del O.A.

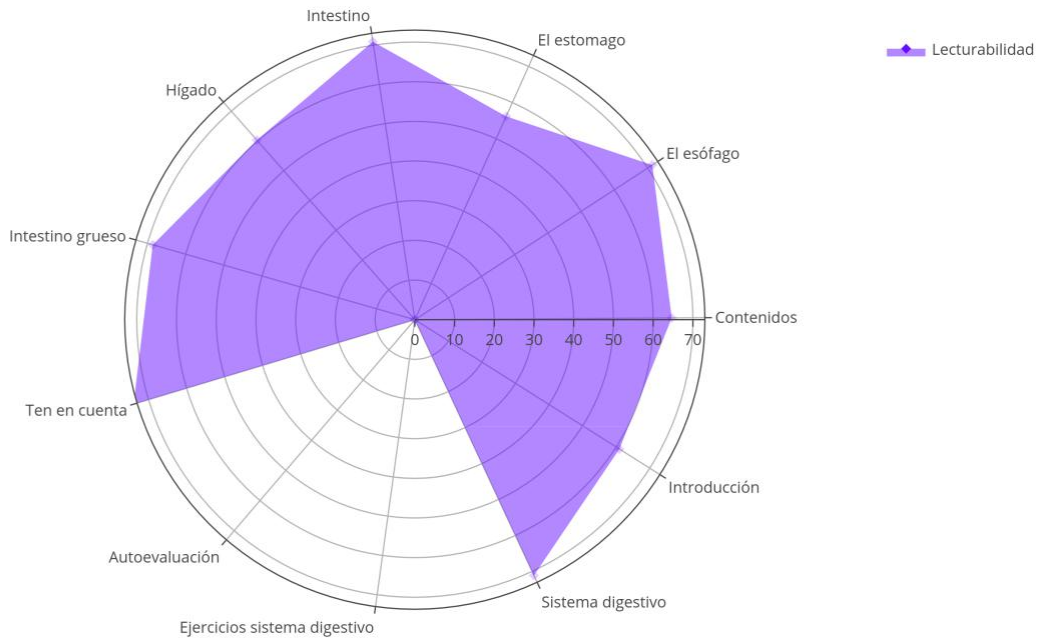


Figura 6.6 Análisis de Lecturabilidad en textos originales extraídos del O.A.

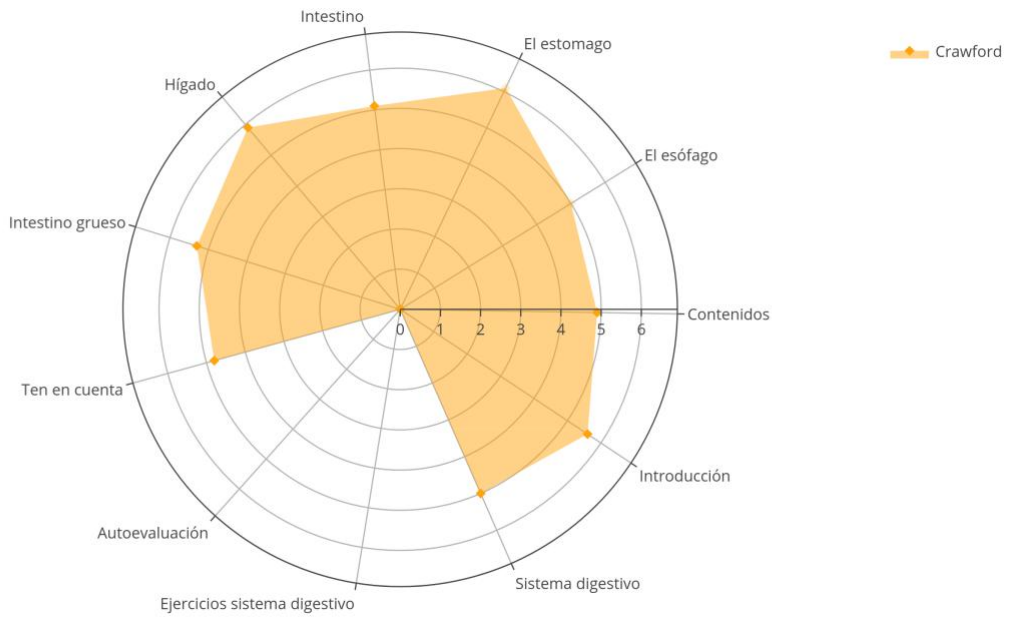


Figura 6.7 Análisis de Crawford en textos originales extraídos del O.A.

A continuación se representa los resultados de los valores calculados por los algoritmos planteados, así como el resultado de los textos procesados por el sistema inteligente y el S.E en el cual se puede identificar que el índice de legibilidad aumenta al emplear dicho sistema inteligente a partir de las recomendaciones dadas, por ejemplo el índice de legibilidad de los textos originales se lo identifica con 62.68 y al emplear el sistema este aumenta a un 66.58, comparándolo con la escala de legibilidad es un índice adecuado.

The screenshot shows a web browser window with the URL localhost:8080/SIO/index.jsf. The page title is "Resultado de procesamiento" and it includes navigation links for "Anterior Página" and "Siguiente Página". The main content is titled "Sistema Digestivo | Sistema Digestivo" and is split into two columns: "Texto Original" and "Texto Procesado".

Under "Texto Original", the text reads: "Todo empieza en la boca Tu sistema digestivo empieza a funcionar incluso antes de se diera el primer bocado a tu pizza. Y seguirá ocupado con la comida que acabas de masticar durante las próximas horas; a veces, hasta días, dependiendo de lo que comas. Este proceso, llamado digestión, permite que el cuerpo obtenga los nutrientes y la energía que necesita de los alimentos que comas. Por lo tanto, averiguemos que le pesa a la pizza, la naranja y la leche. ¿Deseas conocer algo más sobre el sistema digestivo? Aquí encontrarás información muy interesante: enlace."

Under "Texto Procesado", the text reads: "Todo empieza en la boca Tu sistema digestivo empieza a funcionar incluso antes de se diera el primer bocado a tu pizza. Y seguirá ocupado con la comida que acabas de masticar durante las próximas horas. (*) a veces, hasta días, dependiendo de lo que comas. Este proceso, llamado digestión, permite que el cuerpo obtenga los nutrientes y la energía que necesita de los alimentos que comas. entonces (*), averiguemos que le pesa a la pizza, la naranja y la leche. ¿Deseas conocer poco más sobre el sistema digestivo? Aquí encontrarás información muy interesante: enlace."

Below each text sample is a table of readability metrics:

Metric	Original Value	Processed Value
Legibilidad μ	62.68	66.58
Perspicuidad		
Lectura-bilidad		
Análisis Crawford		
Tiempo de lectura		

At the bottom of the page, there is a green button labeled "Guardar Cambios".

Figura 6.8 Ejemplo de presentación de índices de Legibilidad sobre O.A

A continuación se puede presenciar los índices de Legibilidad, Perspicuidad, Lectura-bilidad y análisis Crawford con un crecimiento de índices post procesamiento del sistema inteligente en comparación con los textos originales presentados.

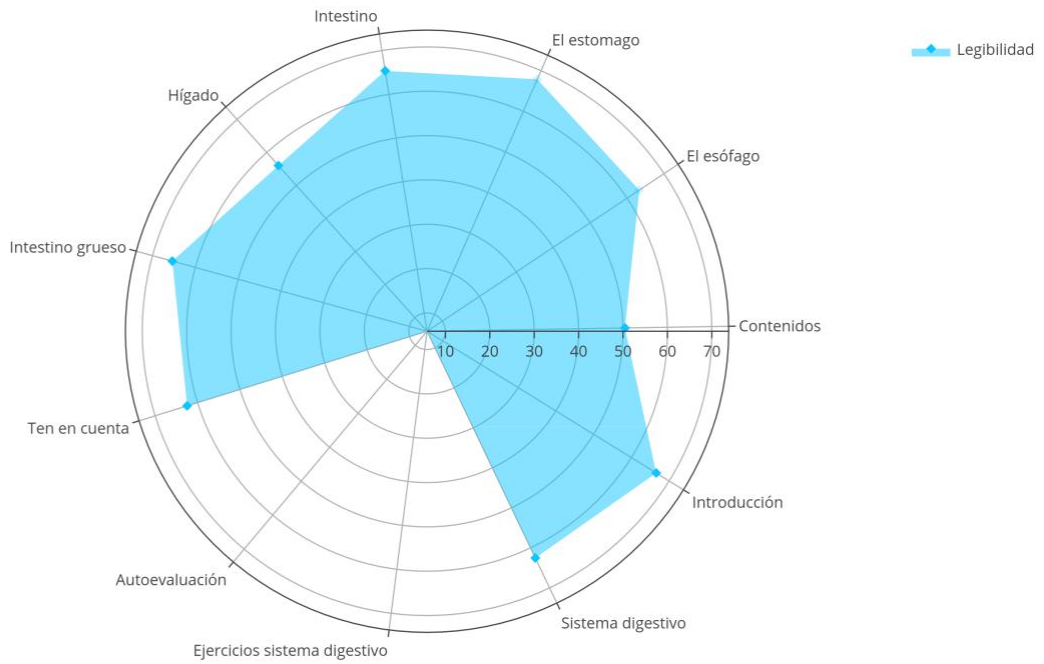


Figura 6.9 Análisis de legibilidad de textos extraídos del O.A. post implementación del procesamiento del sistema inteligente

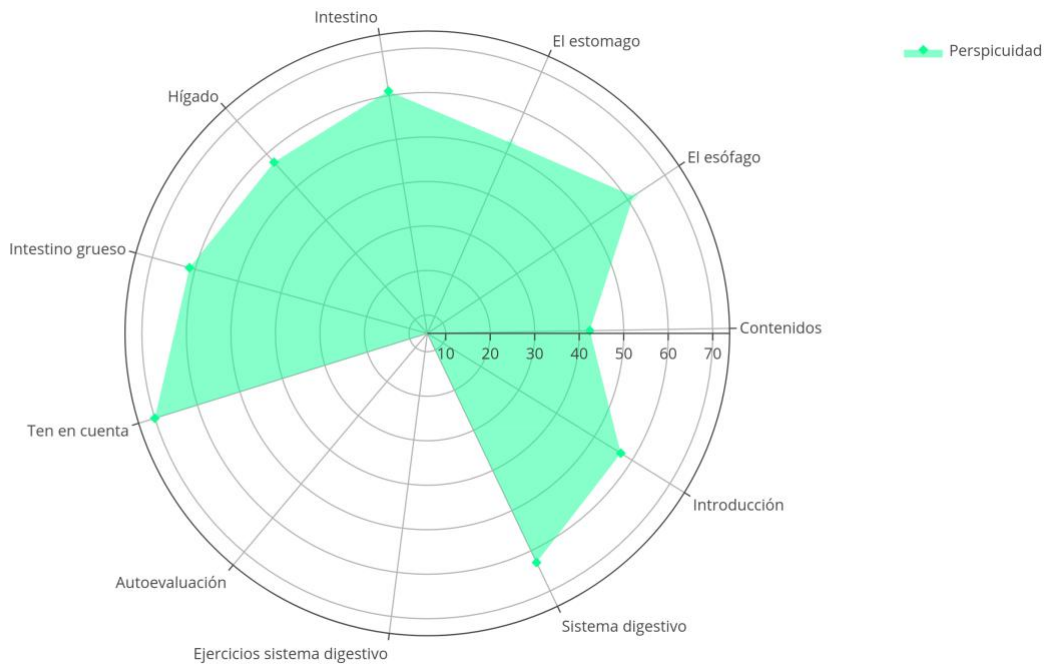


Figura 6.10 Análisis de Perspicuidad de textos extraídos del O.A. post procesamiento del sistema inteligente

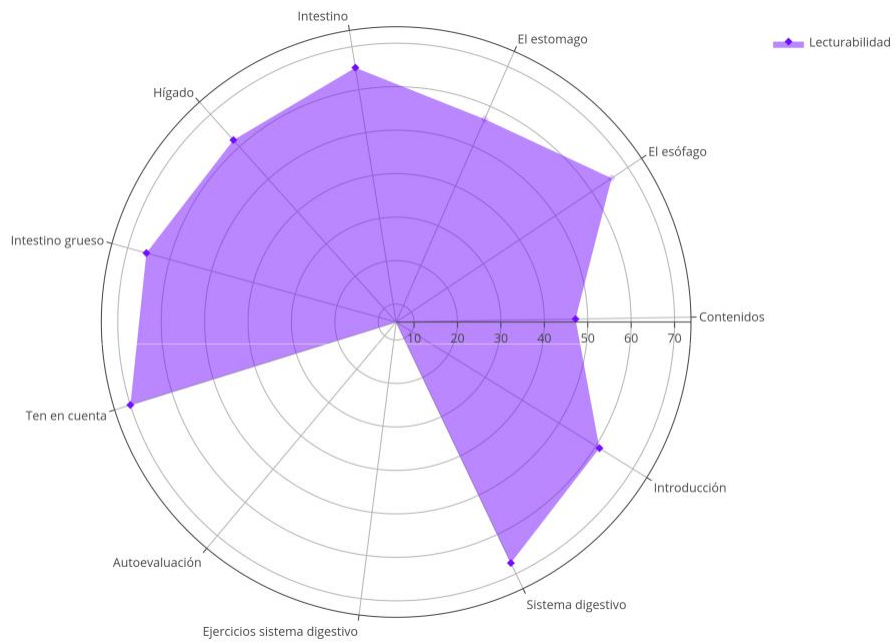


Figura 6.11 Análisis de Lecturabilidad de textos extraídos del O.A. post procesamiento del sistema inteligente

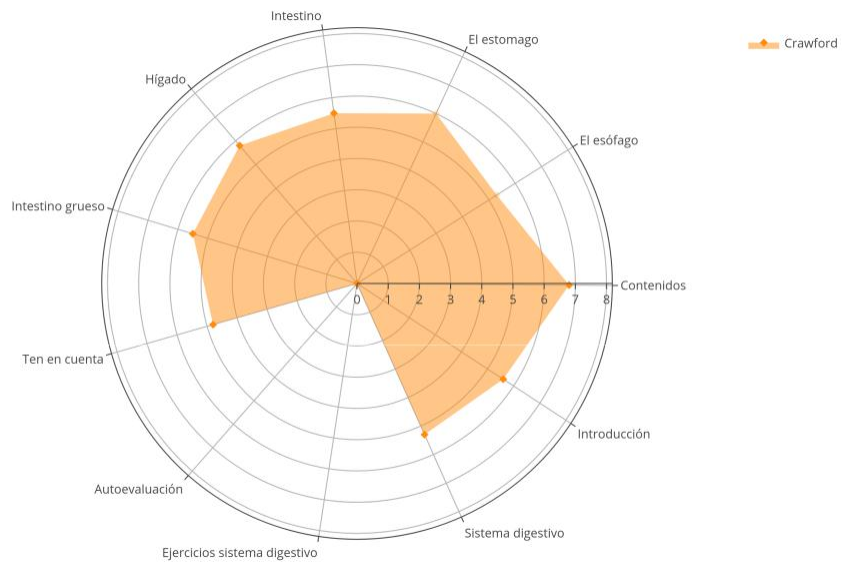


Figura 6.12 Análisis de Crawford de textos extraídos del O.A. post procesamiento del sistema inteligente

A continuación se plantea la experimentación de 30 O.A. divididos en 4 fases correspondientes a los algoritmos de análisis para cada uno de los recursos textuales extraídos de sus componentes HTML. Dicho análisis se lo plantea para los perfiles de discapacidad definidos en el cuadro 6.4.

Experimentación 2 - O.A. Álgebra de Boole : En la figura 6.13 se evidencia el análisis de los textos originales extraídos del O.A. Álgebra de Boole. Y en la figura 6.14 se evidencia el índice de valores postimplementación sobre los textos extraídos.

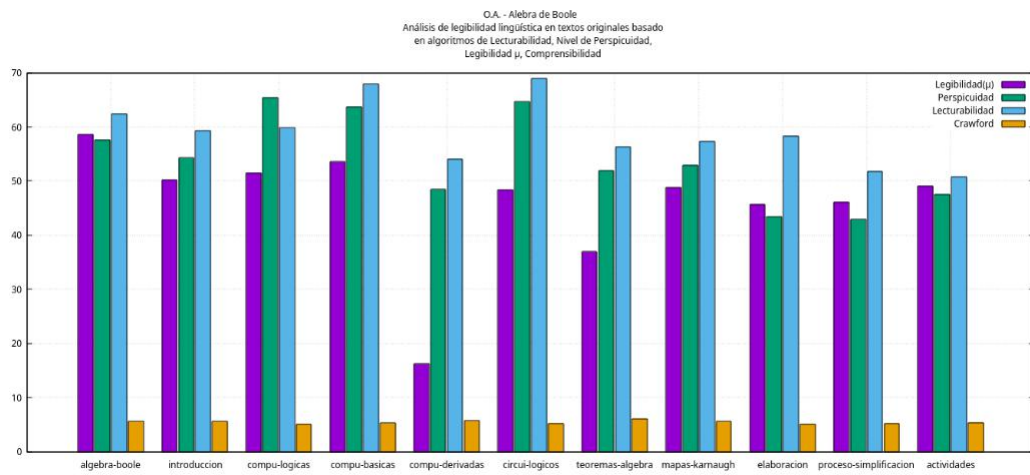


Figura 6.13 Análisis de legibilidad lingüística de textos originales

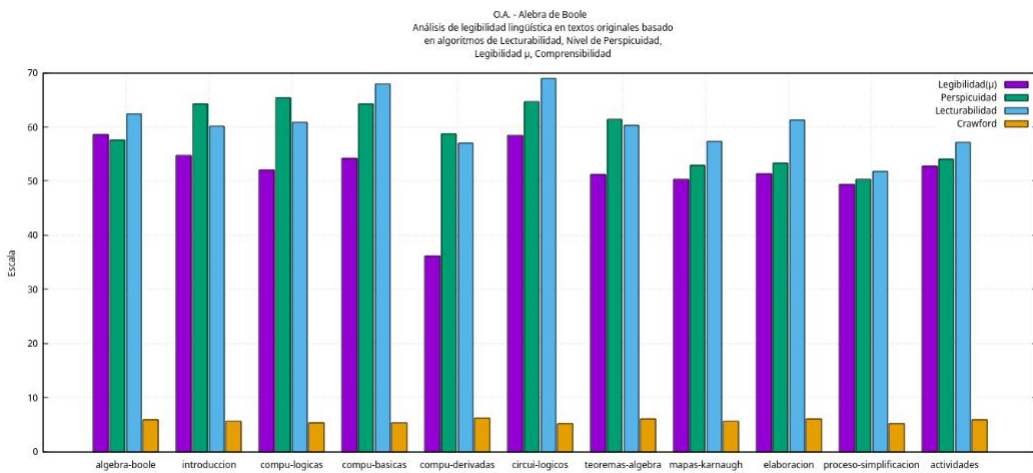


Figura 6.14 Análisis de legibilidad lingüística de textos originales

Experimentación 3 - O.A. Evaluación Primaria del Trauma Infantil [3]: En la figu-ra 6.16 se evidencia el análisis de los textos originales extraídos del O.A. Evaluación Primaria del Trauma Infantil. Y en la figura 6.17 se evidencia el índice de valores postimplementación sobre los textos extraídos.



Figura 6.15 O.A. Evaluación Primaria del Trauma Infantil Procesado

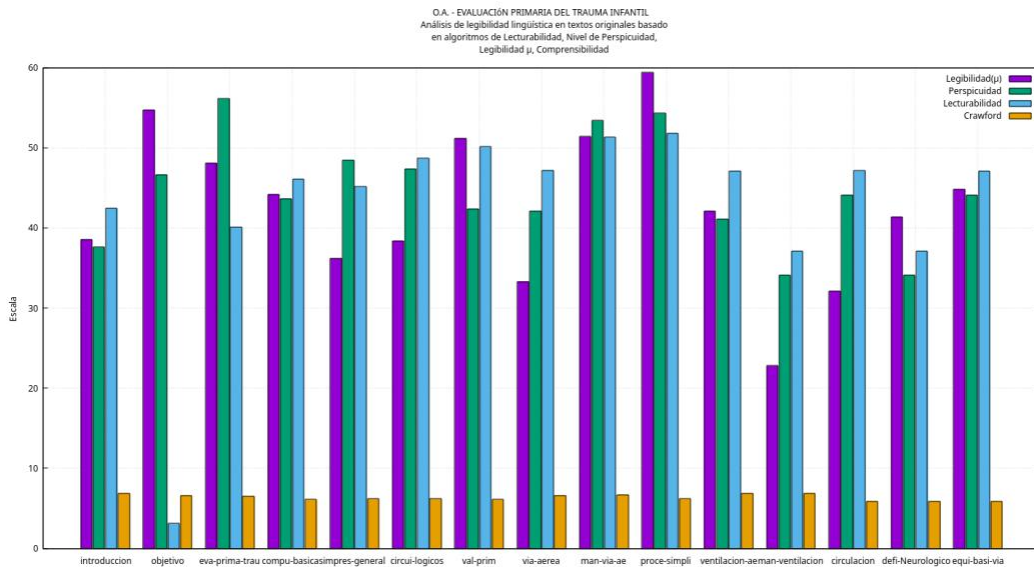


Figura 6.16 Análisis de legibilidad lingüística de textos originales

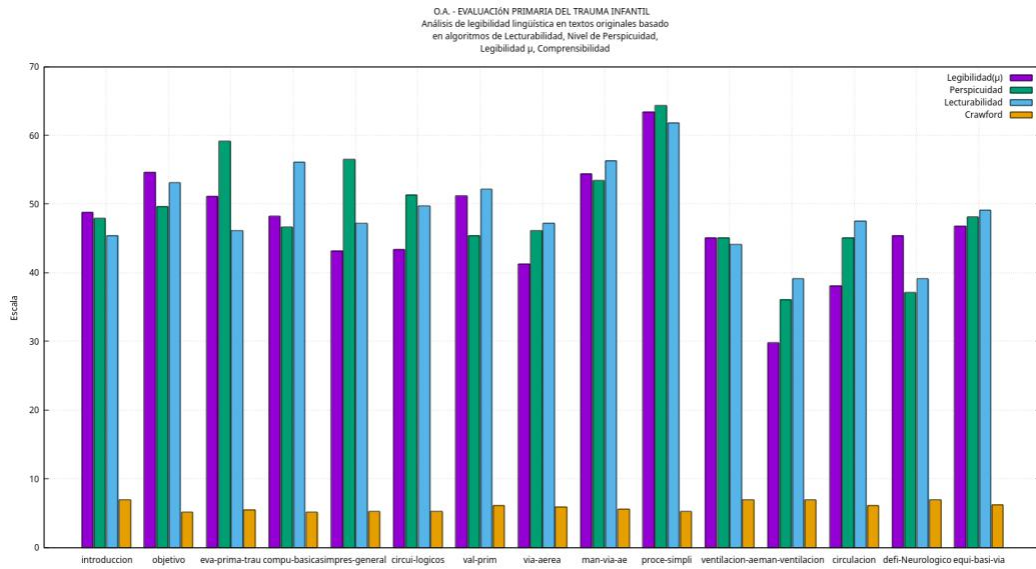


Figura 6.17 Análisis de legibilidad lingüística de textos procesados

Experimentación 4 - O.A. Diseño Instruccional básico para programas formativos virtuales [3]: En la figura 6.19 se evidencia el análisis de los textos originales extraídos del O.A. Evaluación Primaria del Trauma Infantil. Y en la figura 6.20 se evidencia el índice de valores postimplementación sobre los textos extraídos.

¿Qué es diseño instruccional? | Diseño Instruccional básico para programas formativos virtuales

El diseño instruccional es un proceso que parte del análisis o diagnóstico de las necesidades y contexto del cliente, para luego diseñar, desarrollar e implementar un programa formativo haciendo uso de estrategias y técnicas de aprendizaje que convergen con la tecnología para promover un aprendizaje significativo, atractivo, eficaz y eficiente para los estudiantes. El diseño instruccional tiene una importancia relevante en la producción de programas formativos con componentes virtuales, pues se enmarca en el definir las mejores estrategias o alternativas para el aprendizaje considerando el entorno para el cual habrá que diseñar y desarrollar el material didáctico y todos los recursos necesarios, en el contexto de la educación basada en las TIC, si el proceso formativo se desarrolla en un entorno virtual de aprendizaje, será necesario conocer sus capacidades potenciales y sus limitaciones, todo ello conlaja el accionar del diseño instruccional. Recurso Complementario Si desea más información sobre el concepto de diseño instruccional puede descargar el siguiente documento.

Legibilidad μ Perspicuidad Lecturabilidad Análisis Crawford Tiempo de lectura

Incluyen en los cálculos el número de palabras y la media y la varianza del número de letras de las palabras.

valor: 43.3
dificultad: 45.5

Cambiar Cambios

Contrastes procesados

Resultado

Encontrados

Cúlores encontrados: (250, 250, 250): 94.50% (741461 píxeles) (20, 132, 167): 2.22% (17423 píxeles) (9, 9, 9): 1.85% (14555 píxeles) (77, 77, 84): 0.96% (6785 píxeles) (160, 160, 160): 0.60% (4689 píxeles) (255, 5, 5): 0.11% (887 píxeles) (0, 51, 102): 0.05% (410 píxeles) (255, 133, 133): 0.03% (209 píxeles) (146, 94, 73): 0.00% (13 píxeles) Se han procesado: 786432 píxeles de: 786432

Figura 6.18 O.A. Evaluación Primaria del Trauma Infantil Procesado

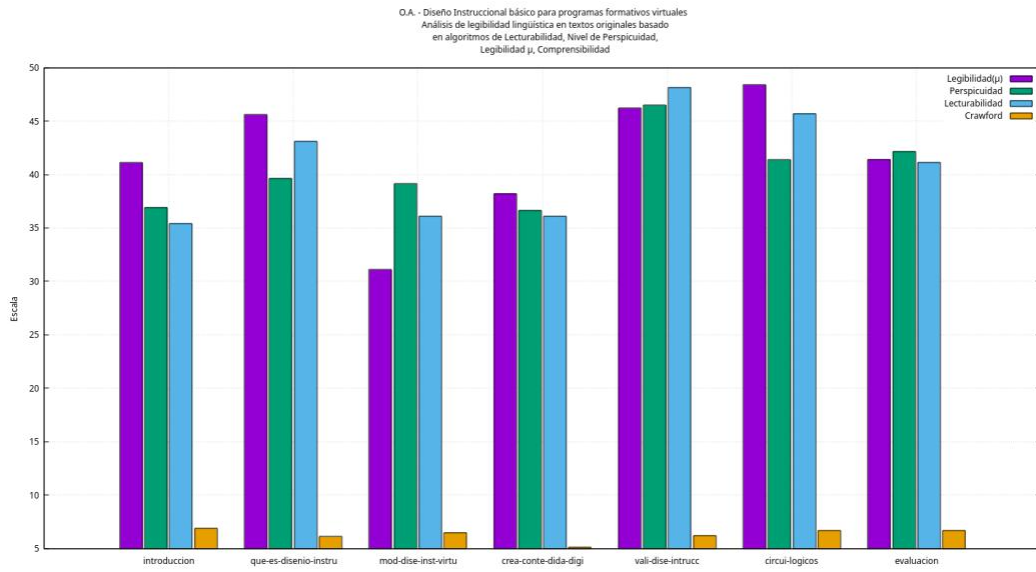


Figura 6.19 Análisis de legibilidad lingüística de textos originales

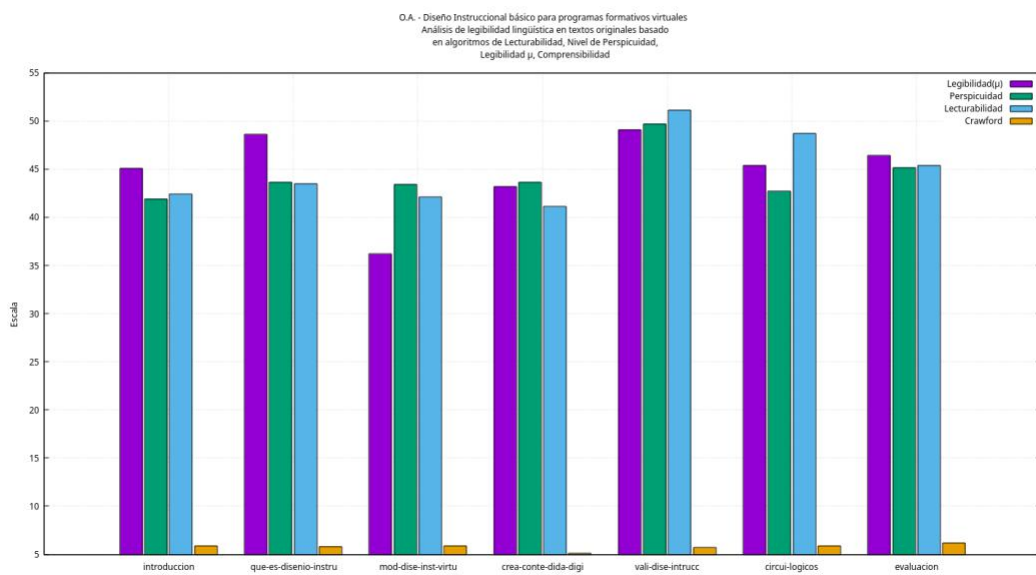


Figura 6.20 Análisis de legibilidad lingüística de textos procesados

Experimentación 5 - O.A. Sistemas numéricos : En la figura 6.22 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.23 se evidencia el índice de valores postimplementación sobre los textos extraídos.

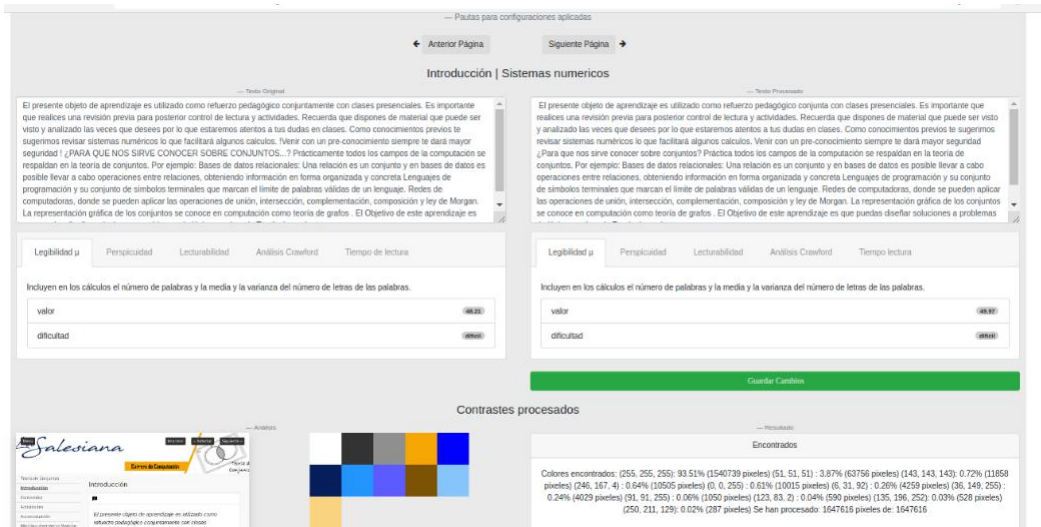


Figura 6.21 O.A. Sistemas numéricos

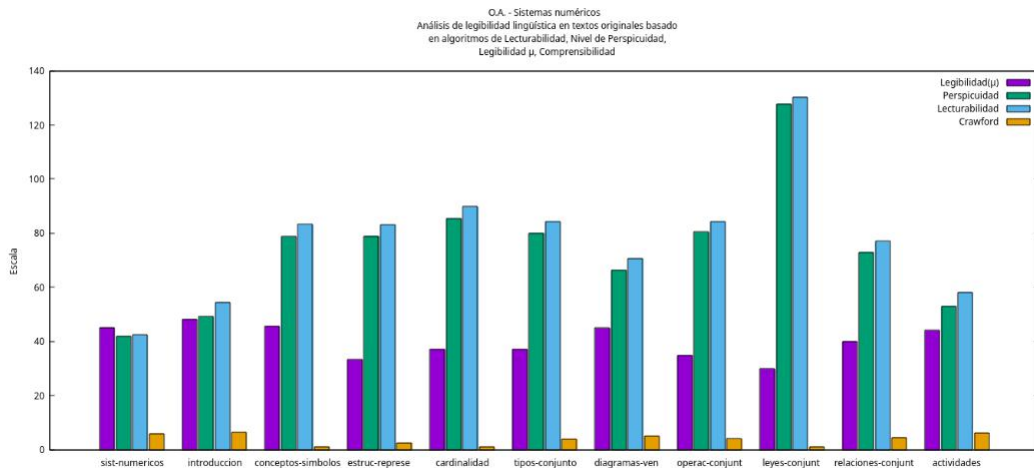


Figura 6.22 Análisis de legibilidad lingüística de textos originales

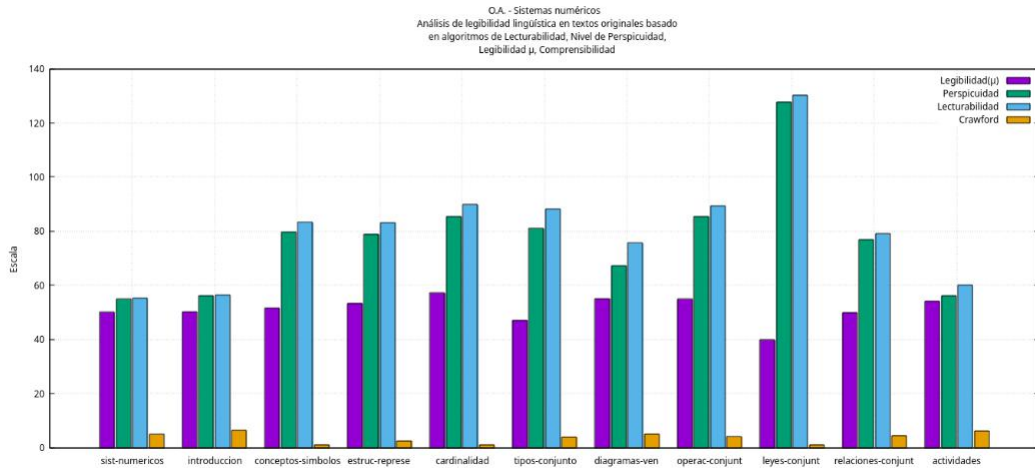


Figura 6.23 Análisis de legibilidad lingüística de textos procesados

Experimentación 6 - O.A. Mitosis [25]: En la figura 6.25 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.26 se evidencia el índice de valores postimplementación sobre los textos extraídos.

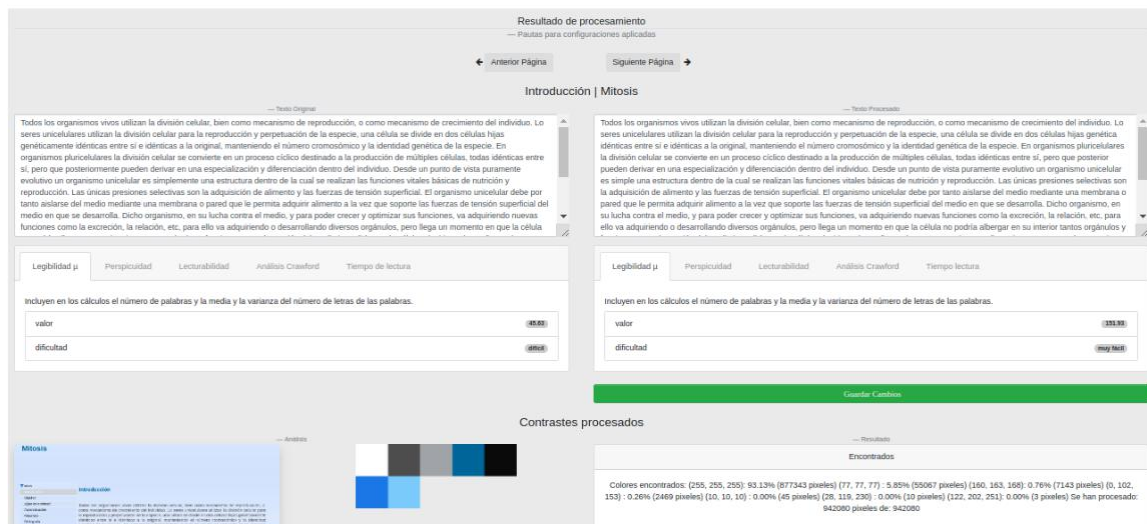


Figura 6.24 O.A. Mitosis

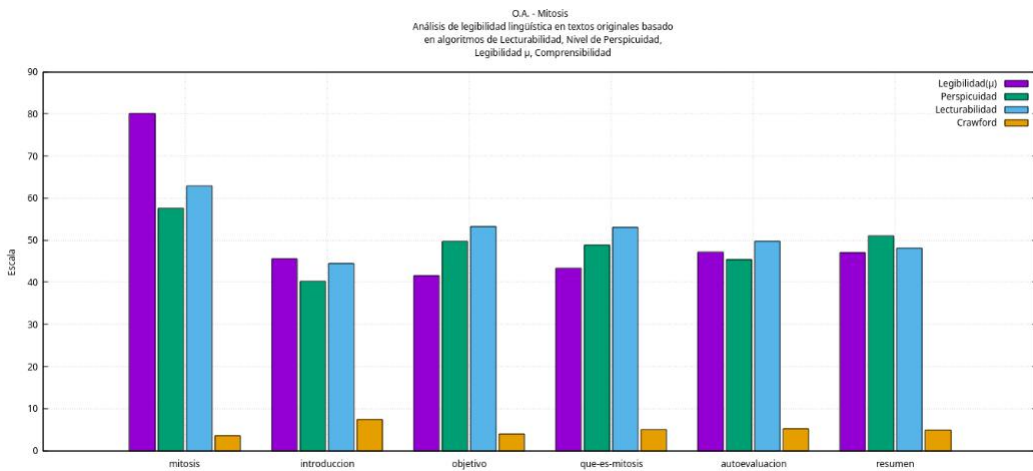


Figura 6.25 Análisis de legibilidad lingüística de textos originales

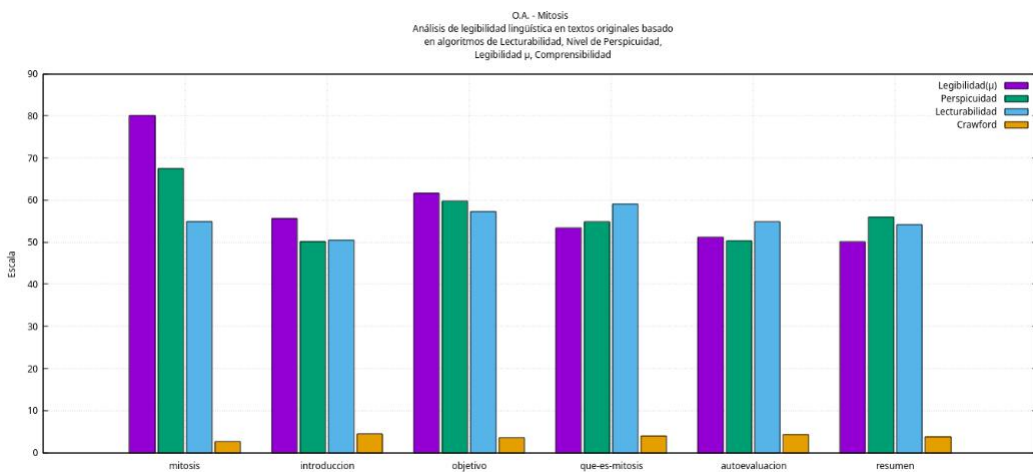


Figura 6.26 Análisis de legibilidad lingüística de textos procesados

Experimentación 7 - O.A. Proceso de la digestión [25]: En la figura 6.28 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.29 se evidencia el índice de valores postimplementación sobre los textos extraídos.



Figura 6.27 O.A. Proceso de la digestión

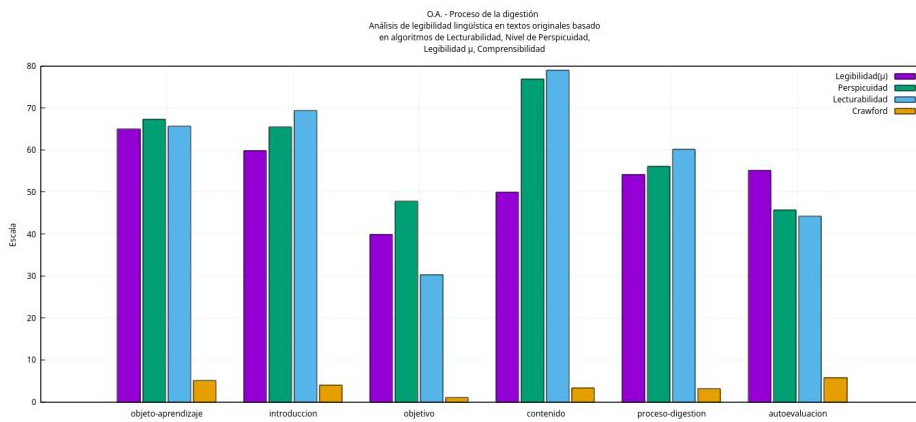


Figura 6.28 Análisis de legibilidad lingüística de textos originales

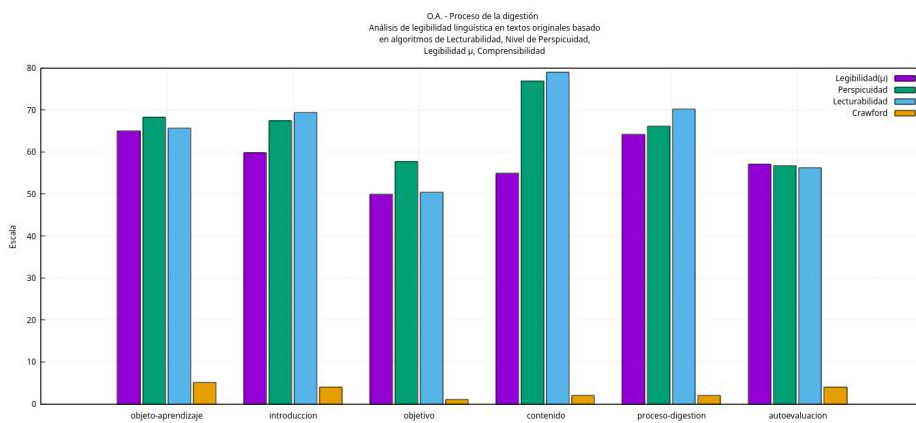


Figura 6.29 Análisis de legibilidad lingüística de textos procesados

Experimentación 8 - O.A. Estructura del núcleo de la célula eucariota [25]: En la figura 6.31 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.32 se evidencia el índice de valores postimplementación sobre los textos extraídos.

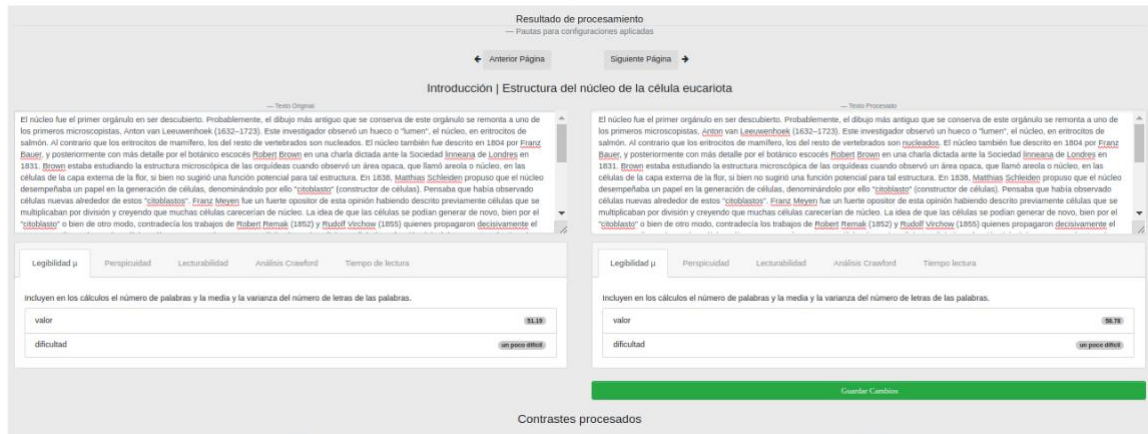


Figura 6.30 O.A. Estructura del núcleo de la célula eucariota

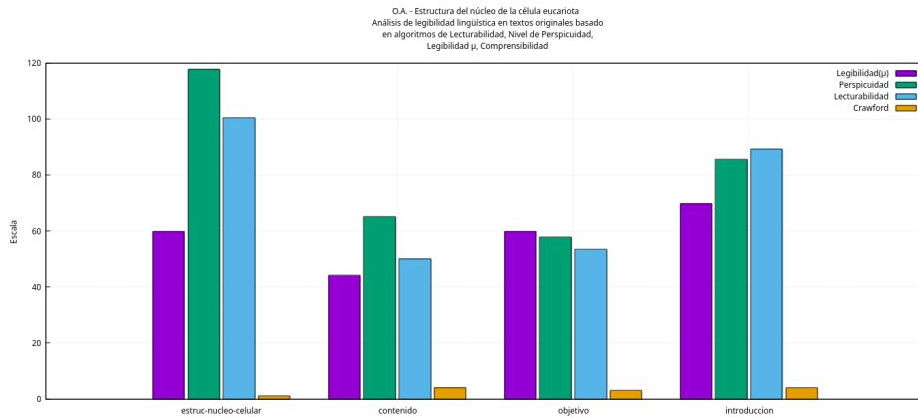


Figura 6.31 Análisis de legibilidad lingüística de textos originales

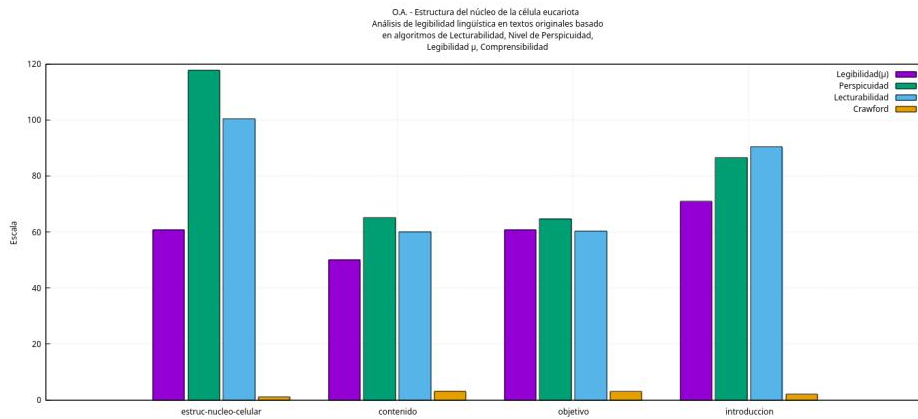


Figura 6.32 Análisis de legibilidad lingüística de textos procesados

Experimentación 9 - O.A. Técnica de medición de la temperatura corporal [25]: En la figura 6.34 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.35 se evidencia el índice de valores postimplementación sobre los textos extraídos.

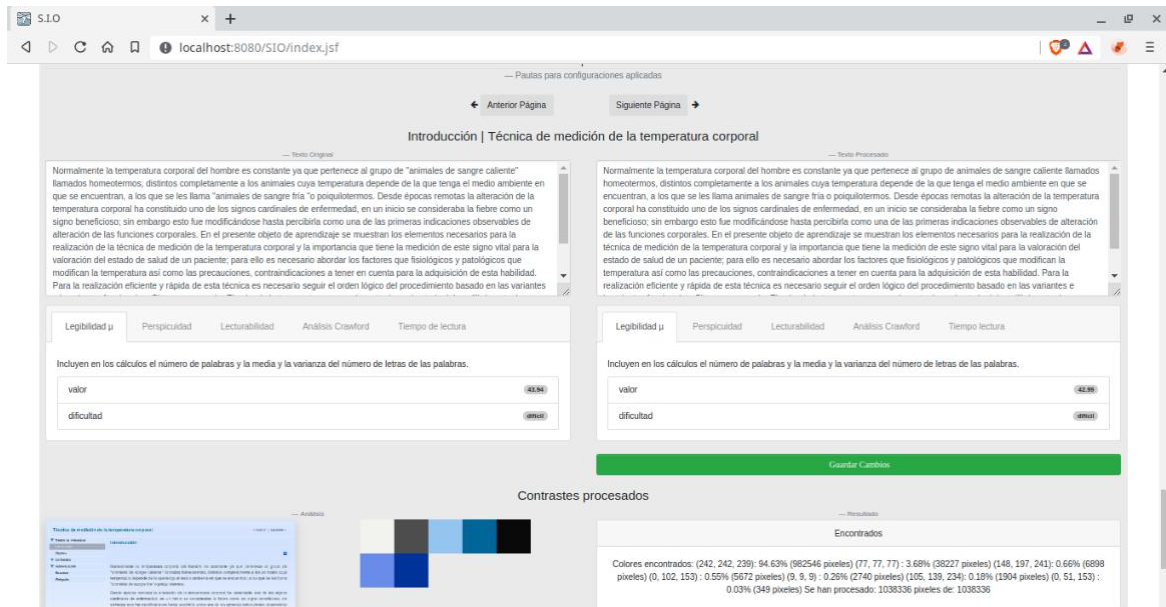


Figura 6.33 O.A. Técnica de medición de la temperatura corporal

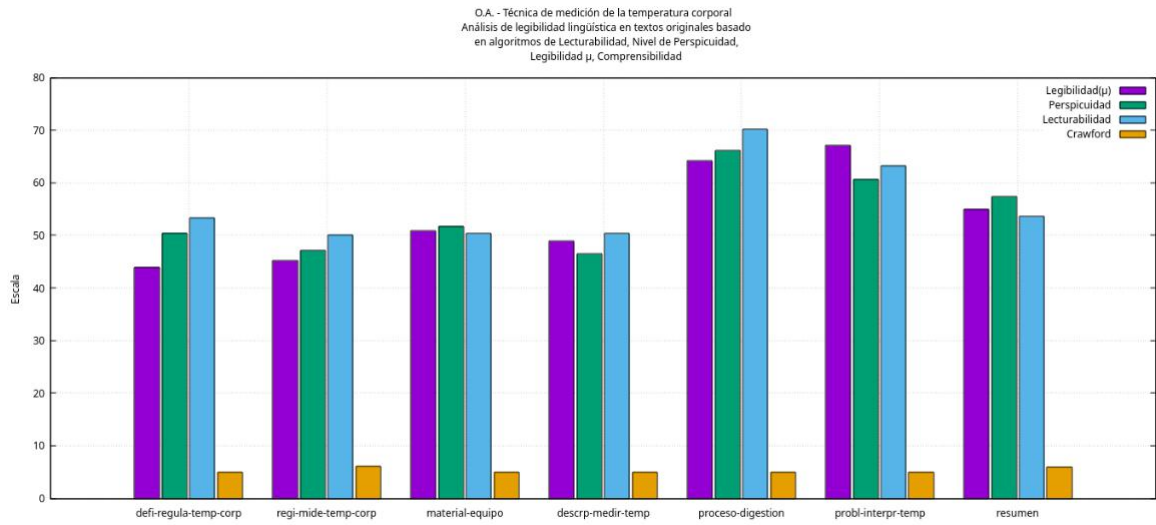


Figura 6.34 Análisis de legibilidad lingüística de textos originales

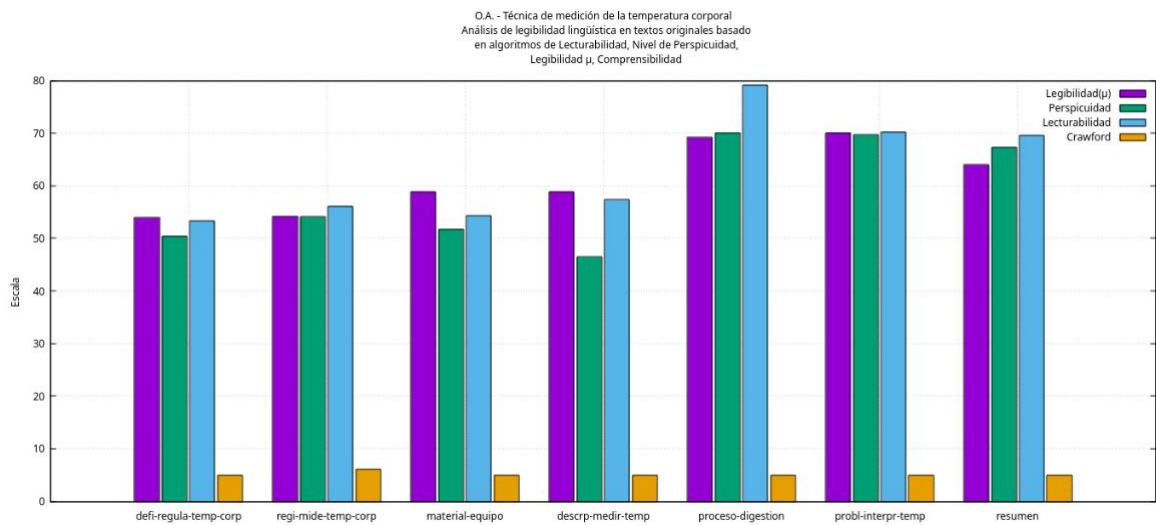


Figura 6.35 Análisis de legibilidad lingüística de textos procesados

Experimentación 10 - O.A. Cromosomas [25]: En la figura 6.37 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.38 se evidencia el índice de valores postimplementación sobre los textos extraídos.

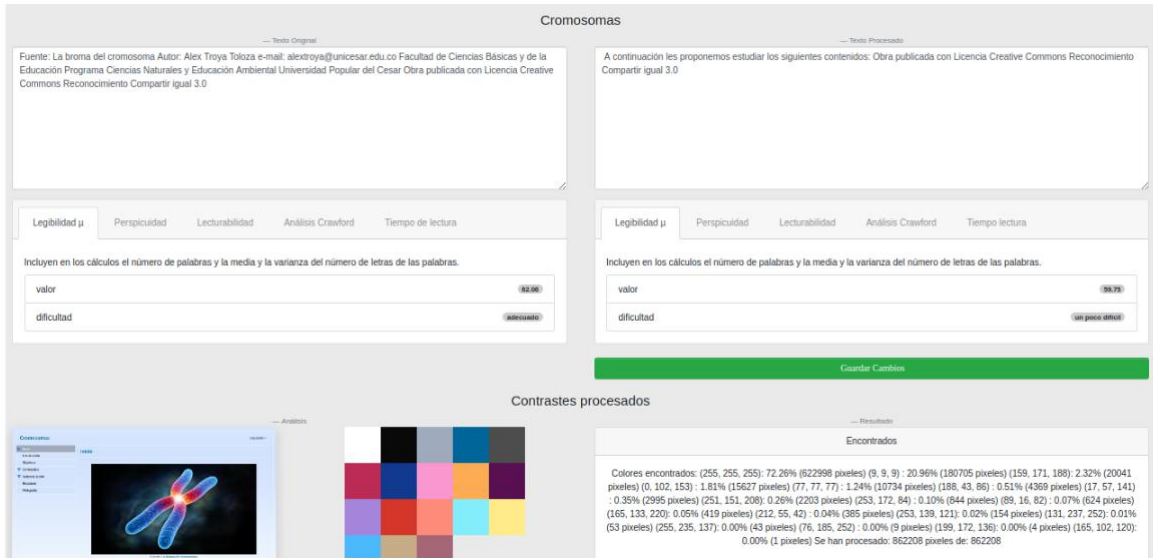


Figura 6.36 O.A. Cromosomas

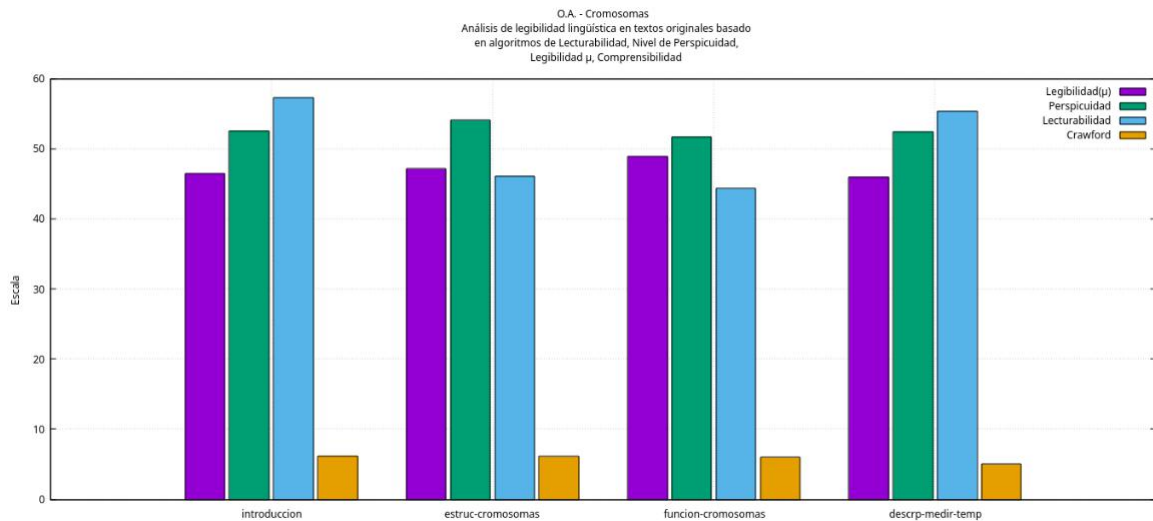


Figura 6.37 Análisis de legibilidad lingüística de textos originales

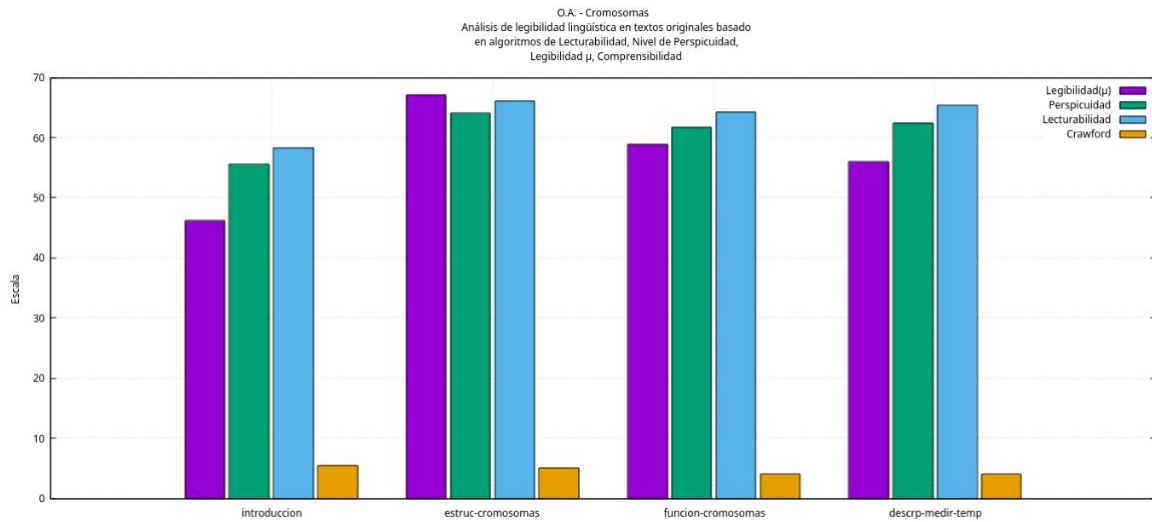


Figura 6.38 Análisis de legibilidad lingüística de textos procesados

Experimentación 11 - O.A. Óxidos [25]: En la figura 6.40 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.41 se evidencia el índice de valores postimplementación sobre los textos extraídos.

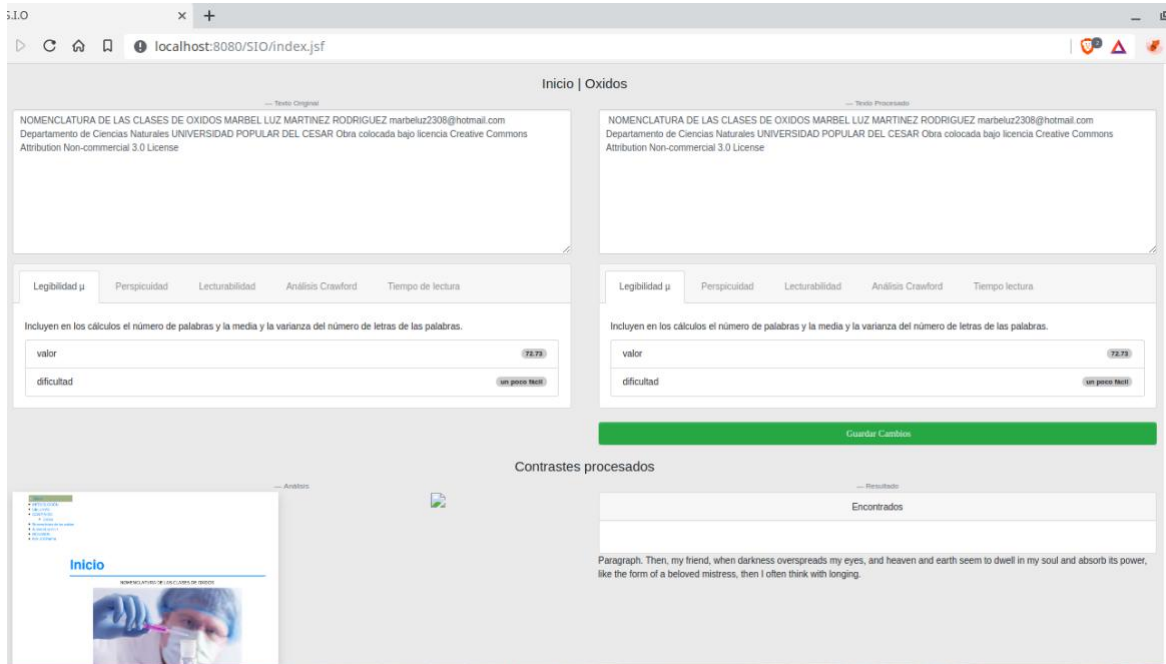


Figura 6.39 O.A. Óxidos

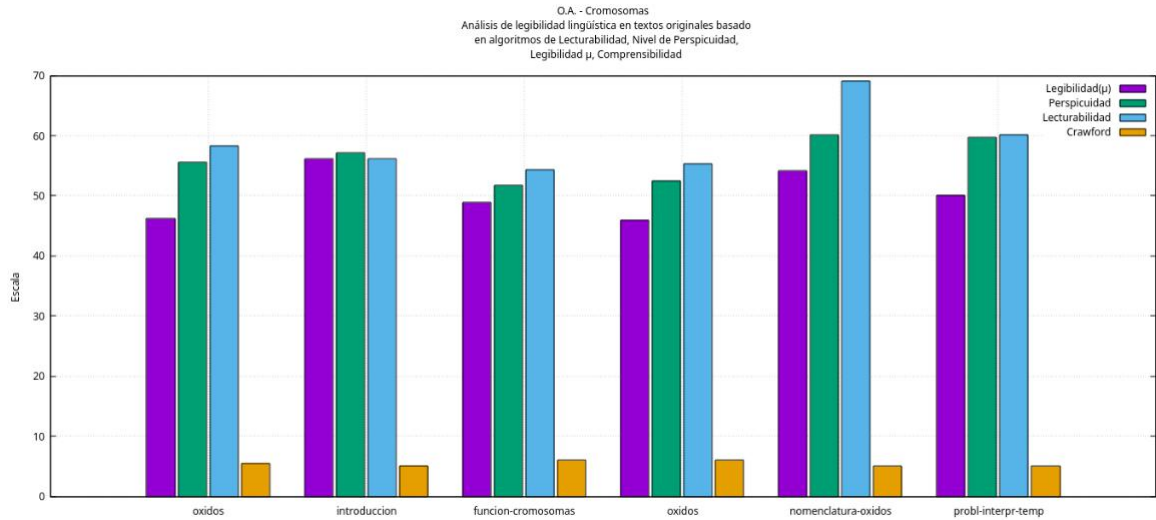


Figura 6.40 Análisis de legibilidad lingüística de textos originales

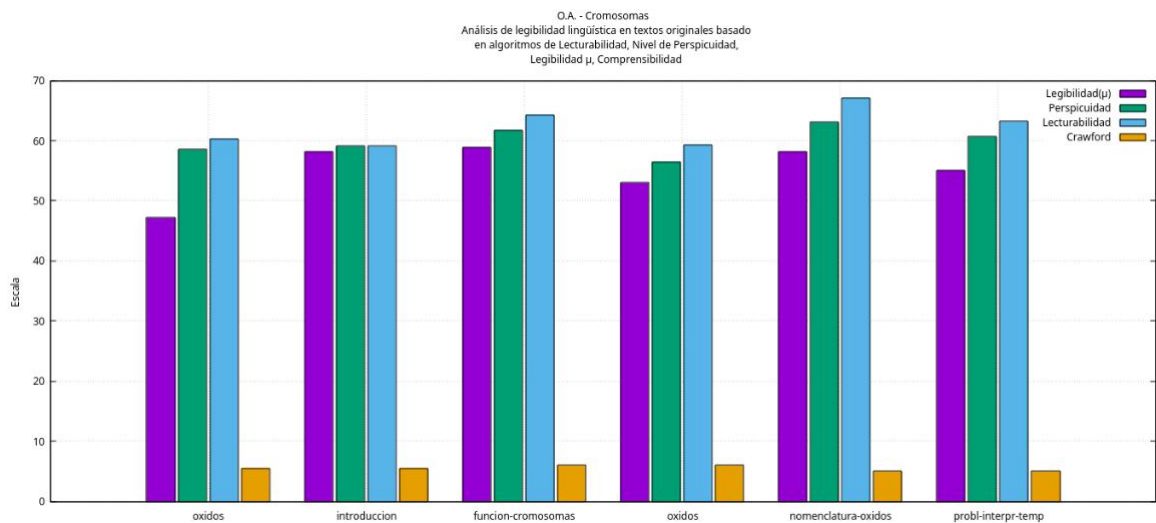


Figura 6.41 Análisis de legibilidad lingüística de textos procesados

Experimentación 12 - O.A. Estructura de un sistema de Telemedicina [3]: En la figura 6.43 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.44 se evidencia el índice de valores postimplementación sobre los textos extraídos.

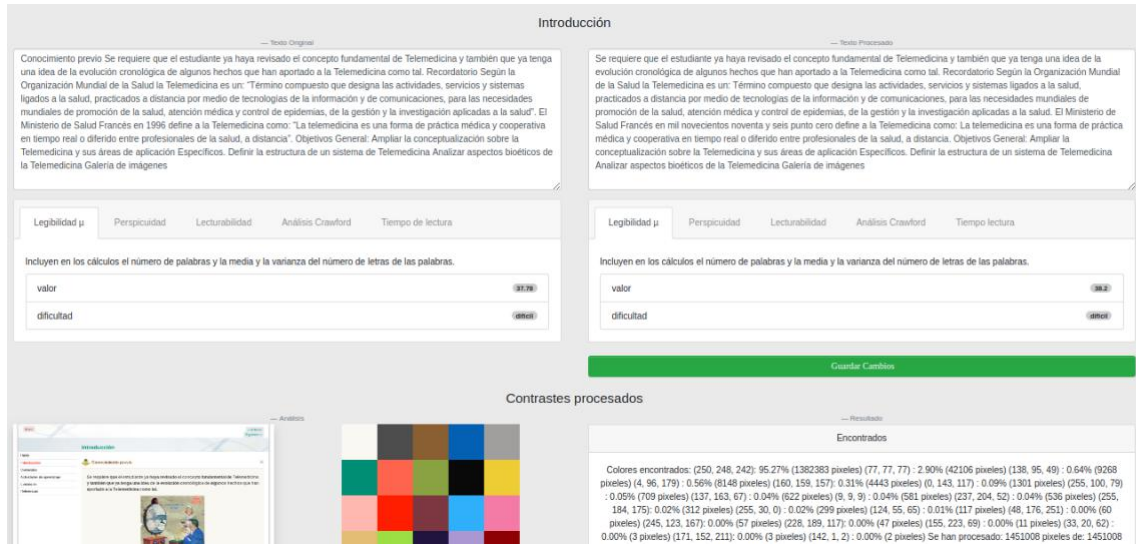


Figura 6.42 O.A. Estructura de un sistema de Telemedicina

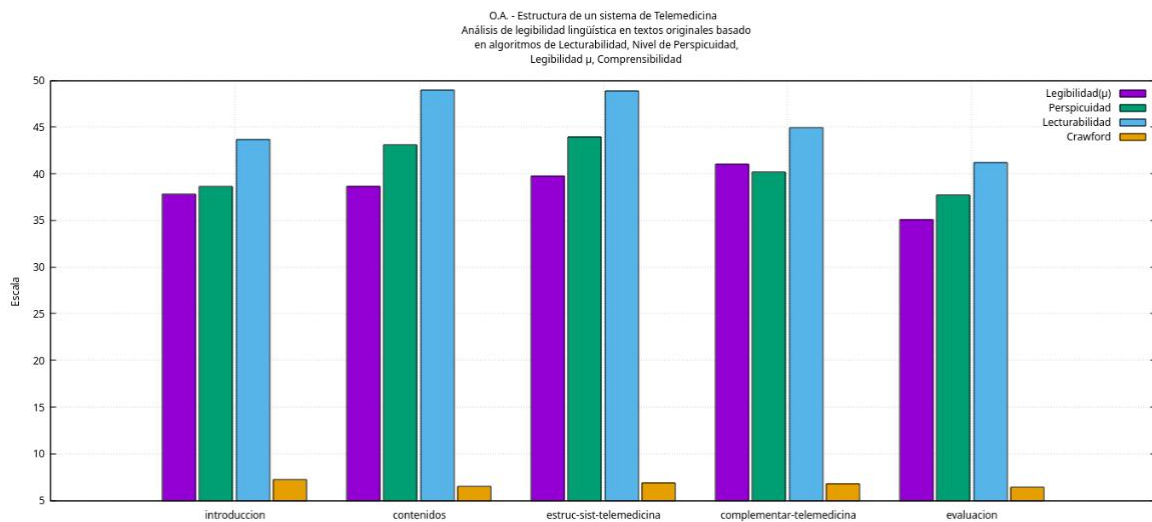


Figura 6.43 Análisis de legibilidad lingüística de textos originales

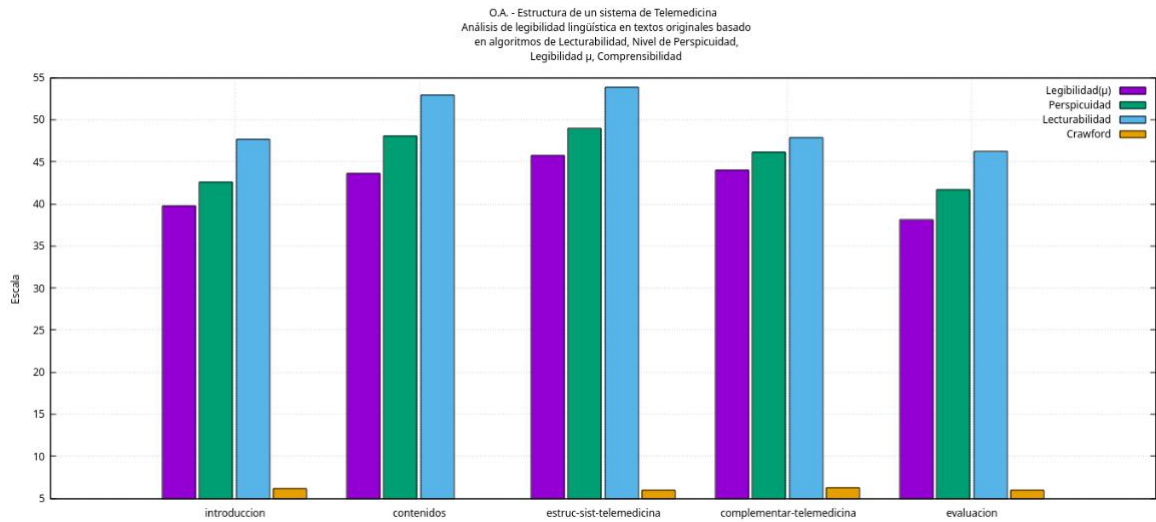


Figura 6.44 Análisis de legibilidad lingüística de textos procesados

Experimentación 13 - O.A. Sistema solar: En la figura 6.46 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.47 se evidencia el índice de valores postimplementación sobre los textos extraídos.

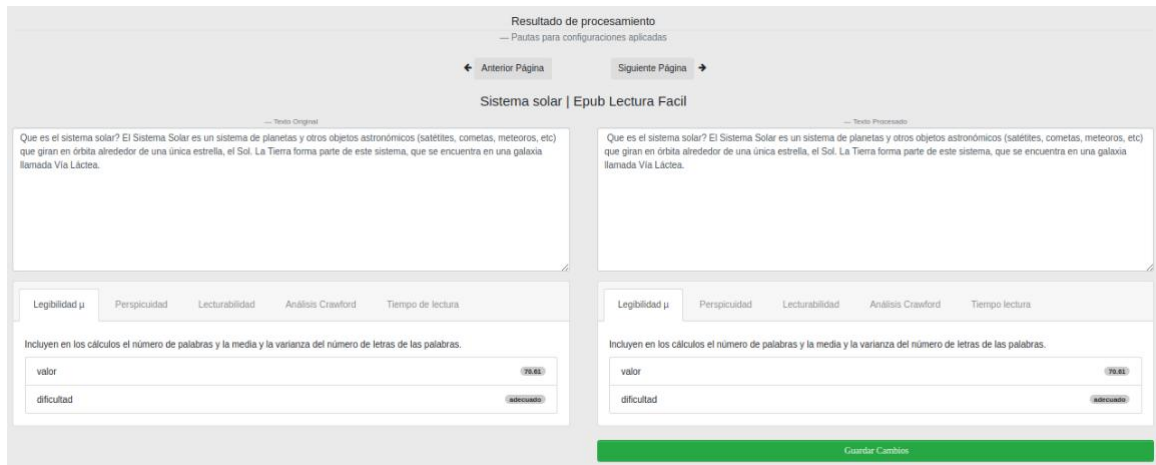


Figura 6.45 O.A. Sistema solar

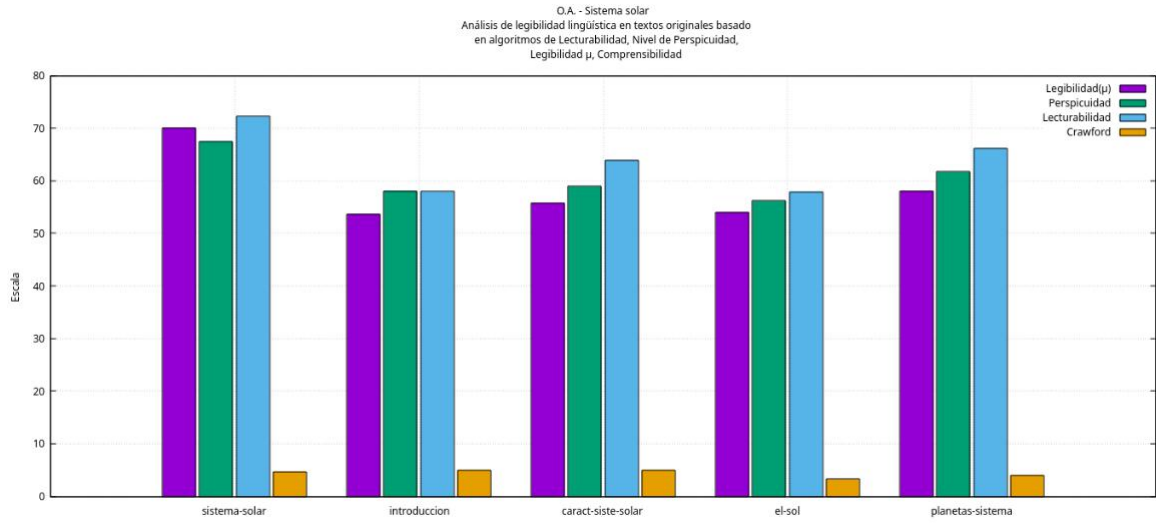


Figura 6.46 Análisis de legibilidad lingüística de textos originales

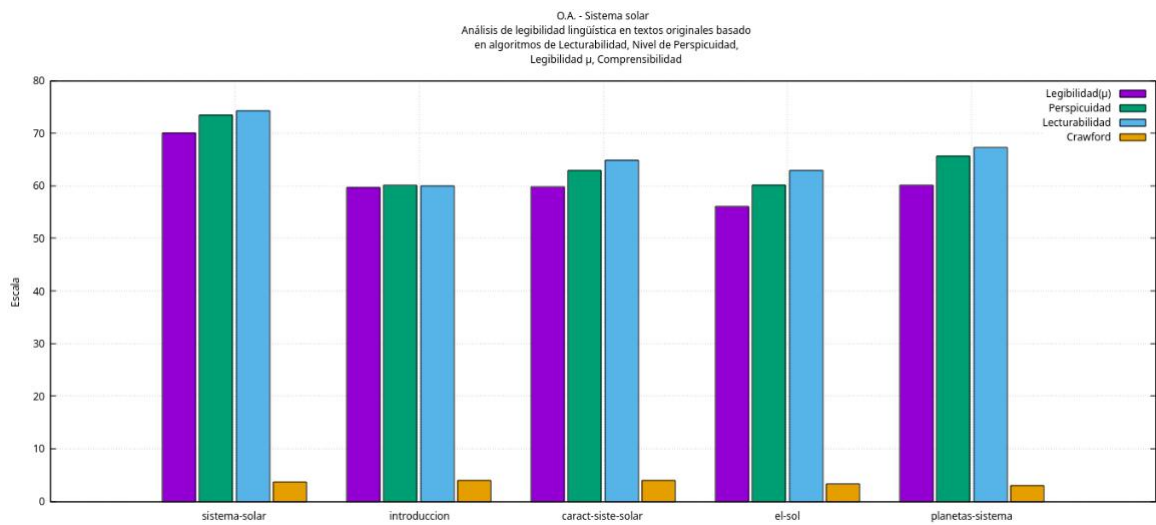


Figura 6.47 Análisis de legibilidad lingüística de textos procesados

Experimentación 14 - O.A. Lógica matemática: En la figura 6.49 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.50 se evidencia el índice de valores postimplementación sobre los textos extraídos.

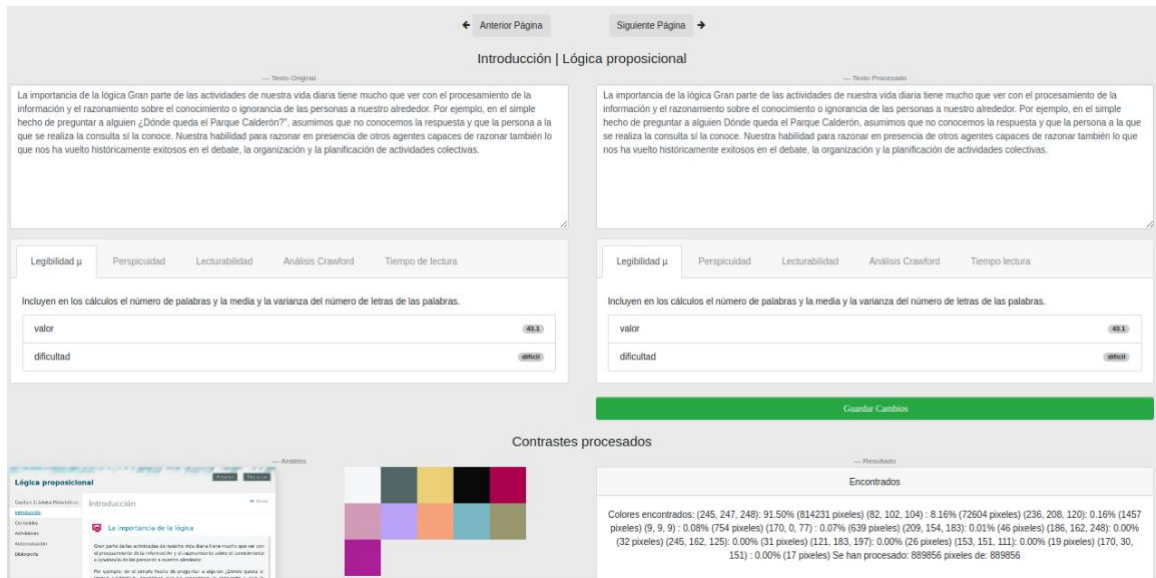


Figura 6.48 O.A. Lógica matemática

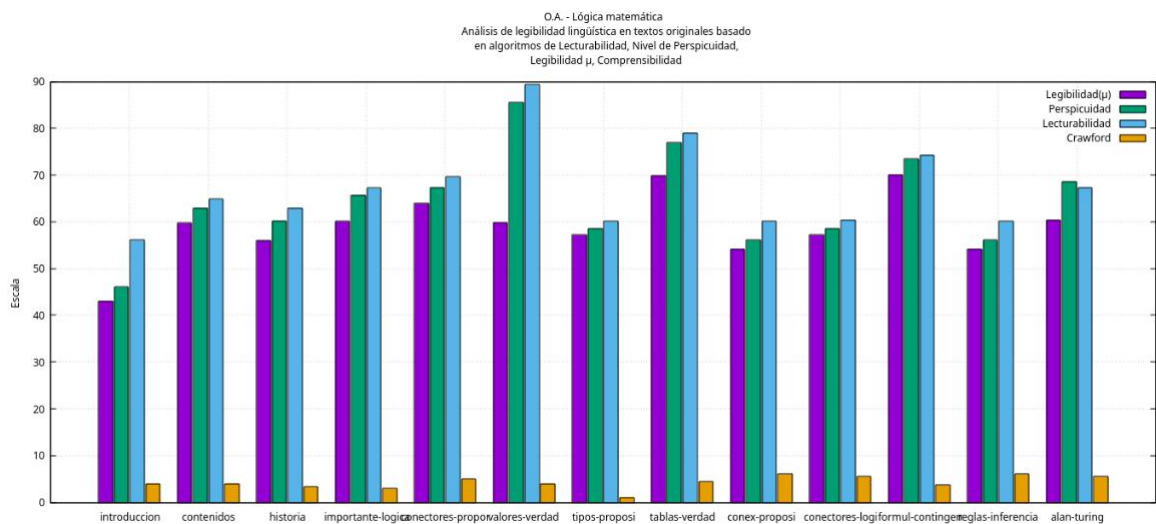


Figura 6.49 Análisis de legibilidad lingüística de textos originales

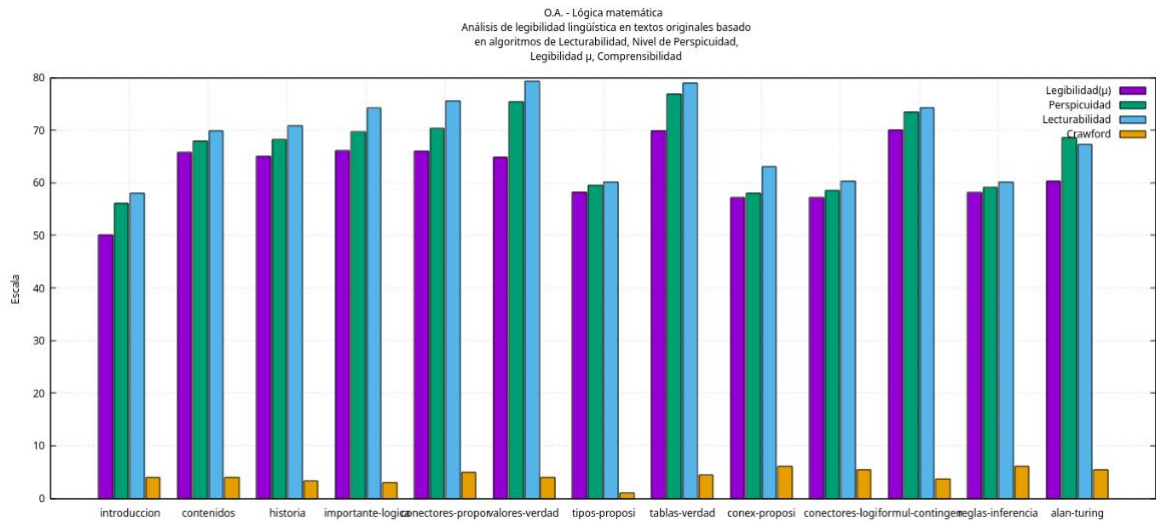


Figura 6.50 Análisis de legibilidad lingüística de textos procesados

Experimentación 15 - O.A. Algoritmos: En la figura 6.52 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.53 se evidencia el índice de valores postimplementación sobre los textos extraídos.

Contrastes procesados

Resultados

Encontrados

Colores encontrados: (255, 255, 255): 92.40% (1325129 pixeles) (82, 102, 104) : 7.27% (104286 pixeles) (232, 182, 21) : 0.10% (1385 pixeles) (208, 188, 117): 0.08% (1061 pixeles) (170, 0, 77) : 0.07% (1061 pixeles) (9, 9, 9) : 0.05% (754 pixeles) (125, 197, 211) : 0.01% (189 pixeles) (209, 154, 183): 0.01% (75 pixeles) (170, 30, 151) : 0.00% (44 pixeles) (186, 162, 248): 0.00% (39 pixeles) (235, 154, 125): 0.00% (17 pixeles) Se han procesado: 1434060 pixeles de: 1434060

Figura 6.51 O.A. Algoritmos

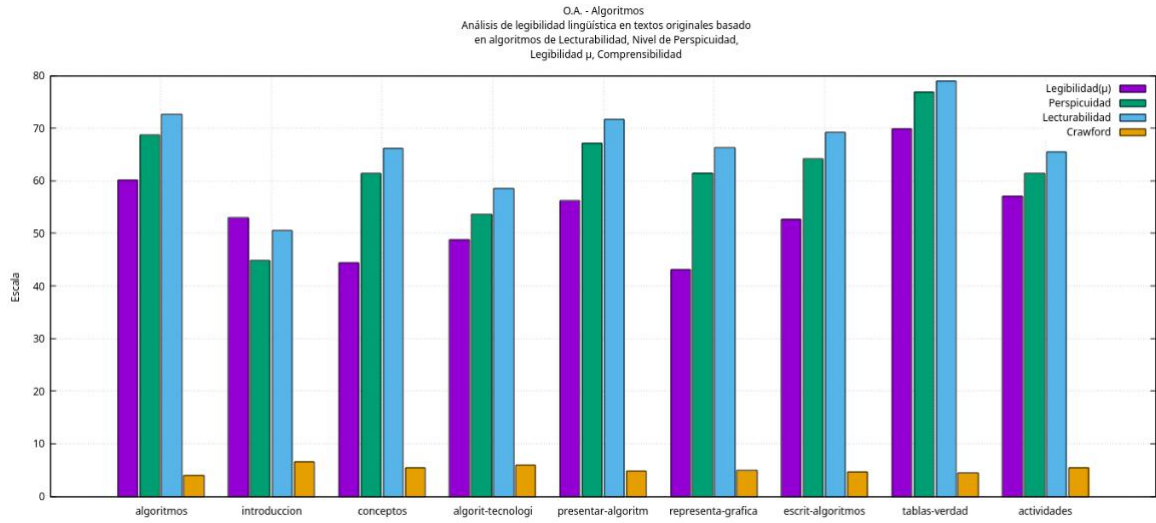


Figura 6.52 Análisis de legibilidad lingüística de textos originales

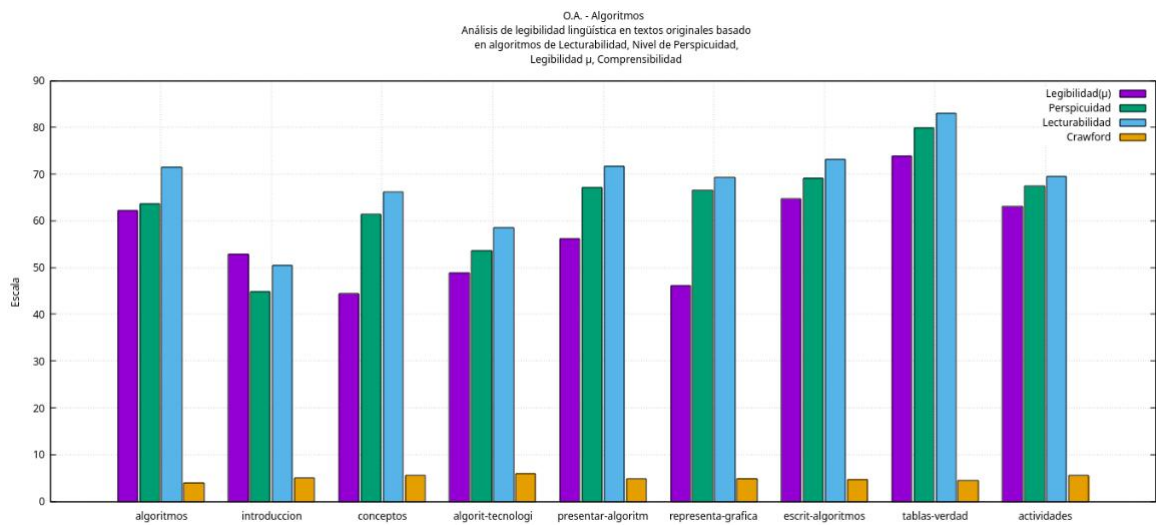


Figura 6.53 Análisis de legibilidad lingüística de textos procesados

Experimentación 16 - O.A. Teoría de conjuntos: En la figura 6.55 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.56 se evidencia el índice de valores postimplementación sobre los textos extraídos.

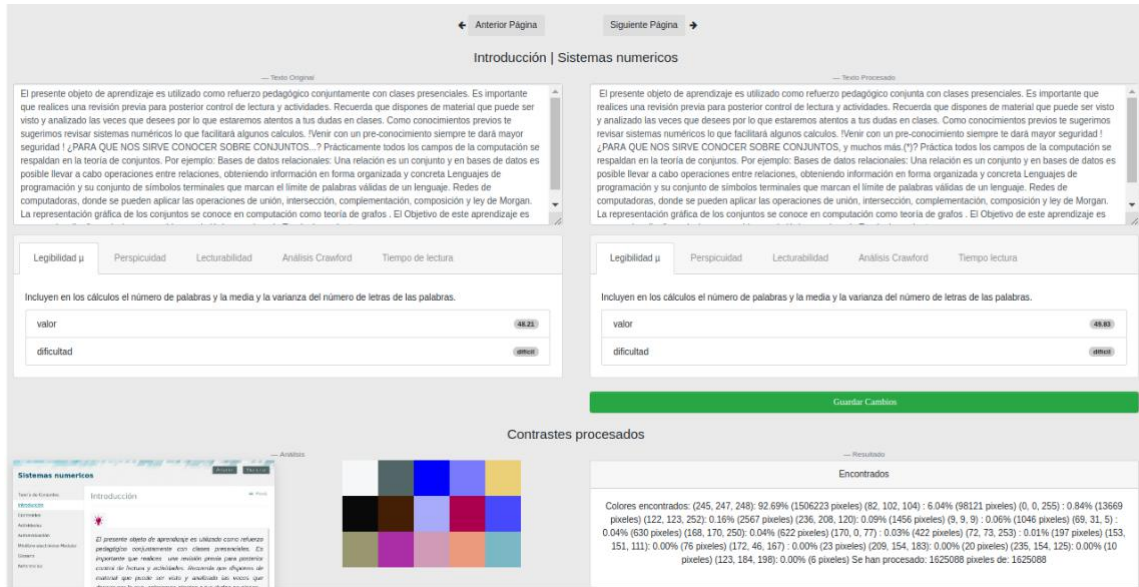


Figura 6.54 O.A. Teoría de conjuntos

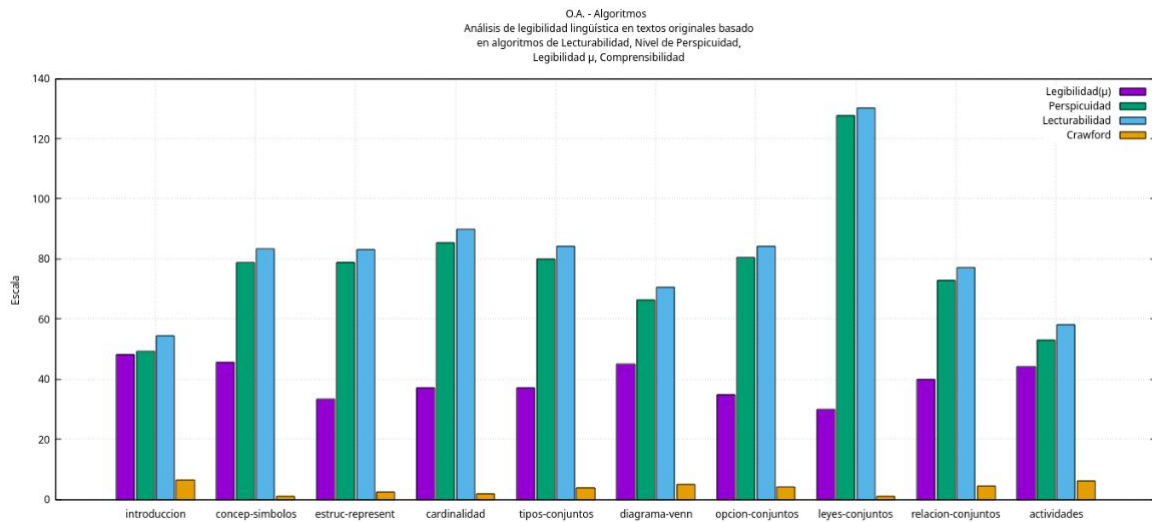


Figura 6.55 Análisis de legibilidad lingüística de textos originales

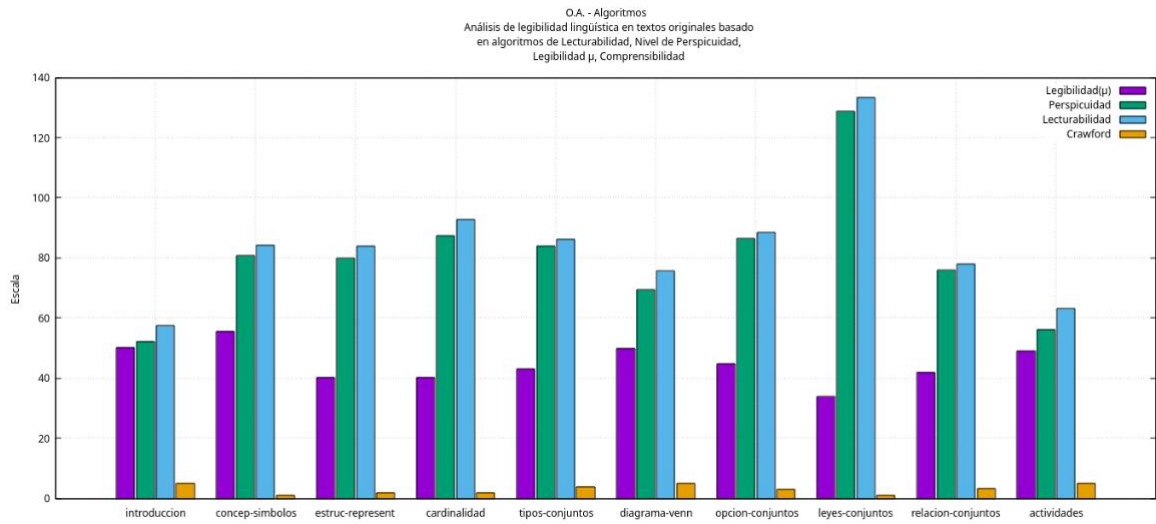


Figura 6.56 Análisis de legibilidad lingüística de textos procesados

Experimentación 17 - O.A. Organización y Arquitectura del Computador: En la figura 6.58 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.59 se evidencia el índice de valores postimplementación sobre los textos extraídos.

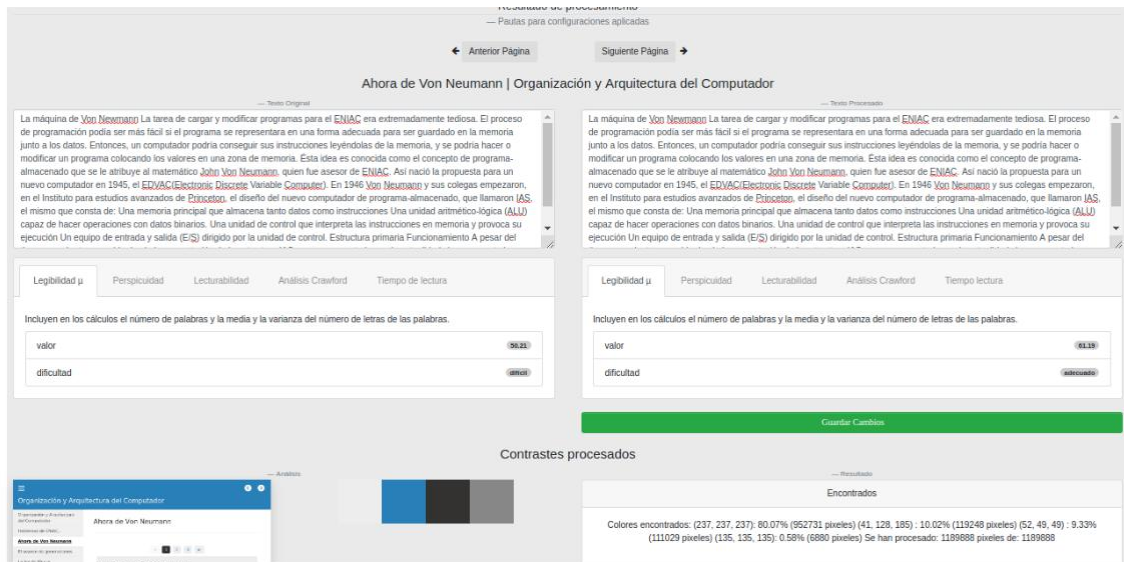


Figura 6.57 O.A. Organización y Arquitectura del Computador

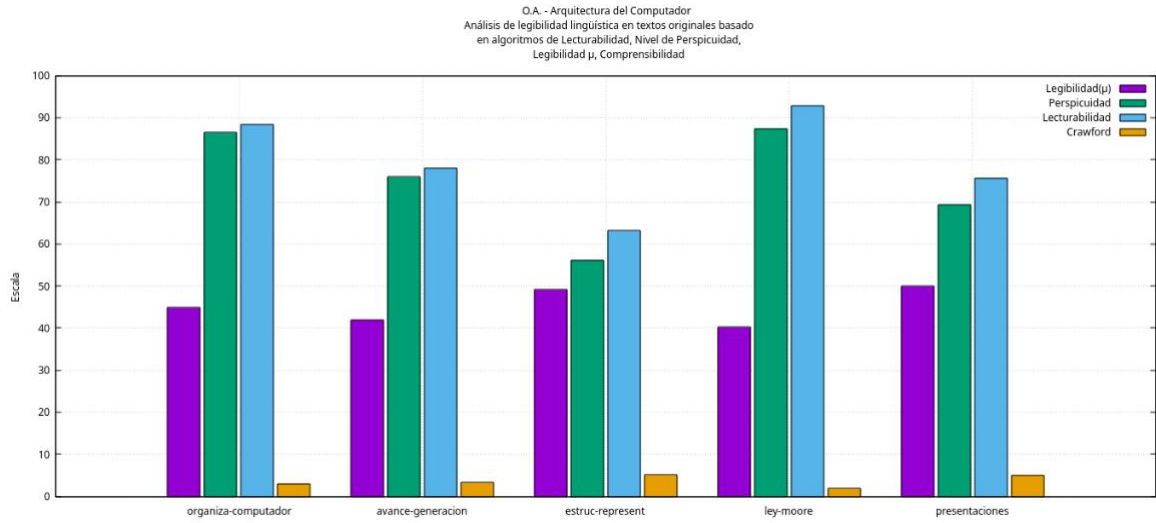


Figura 6.58 Análisis de legibilidad lingüística de textos originales

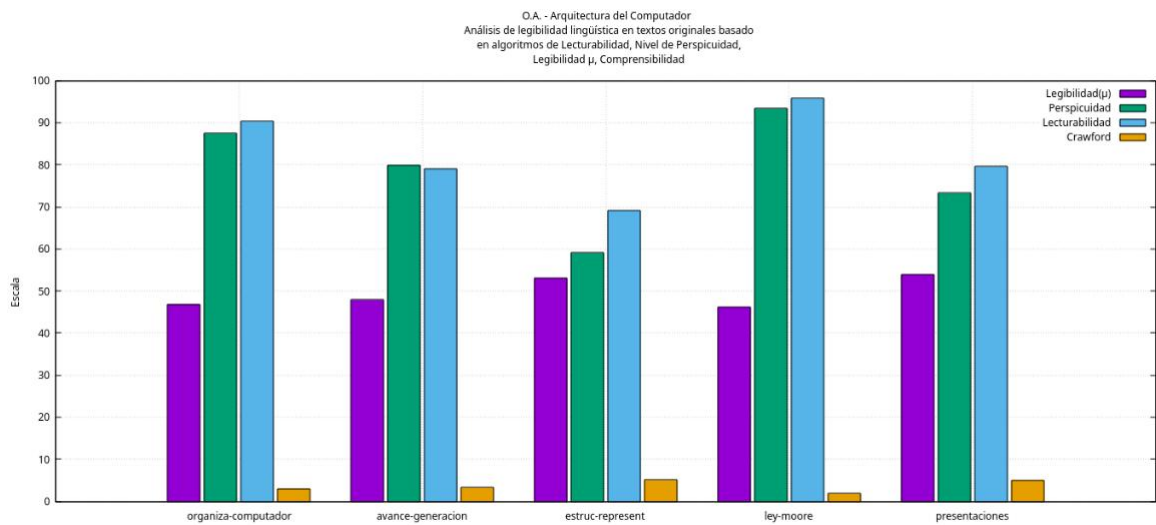


Figura 6.59 Análisis de legibilidad lingüística de textos procesados

Experimentación 18 - O.A. Introducción Salud sexual y reproductiva: En la figura 6.61 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.62 se evidencia el índice de valores postimplementación sobre los textos extraídos.

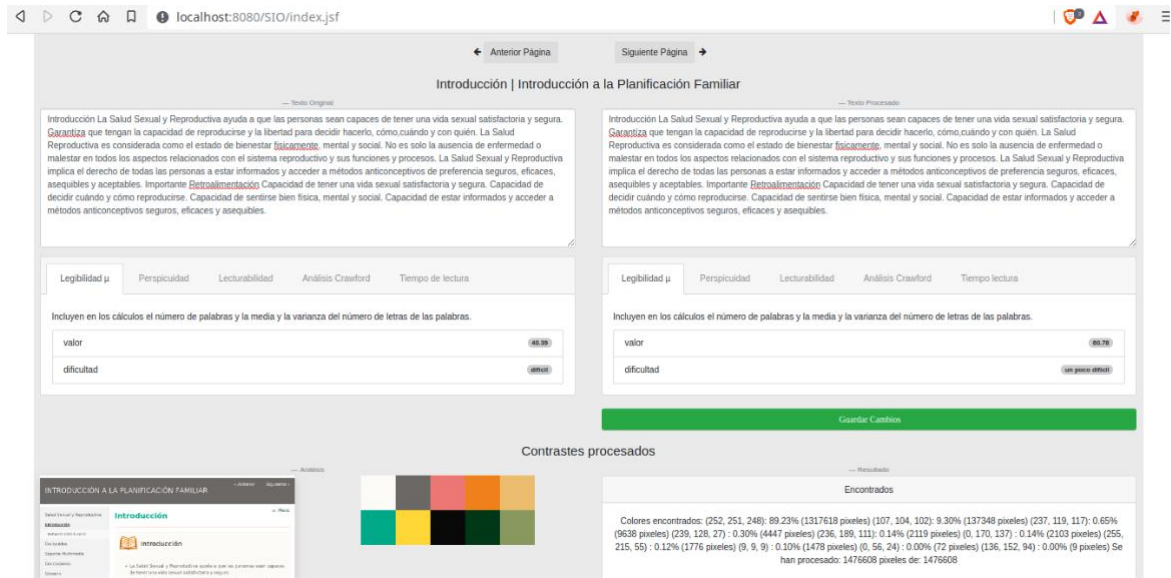


Figura 6.60 O.A. Introducción Salud sexual y reproductiva

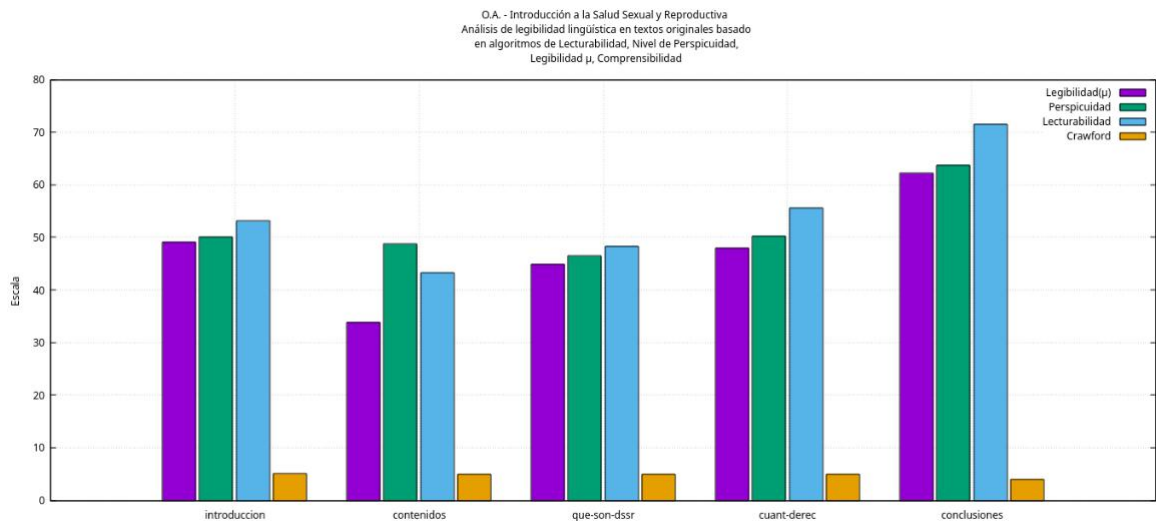


Figura 6.61 Análisis de legibilidad lingüística de textos originales

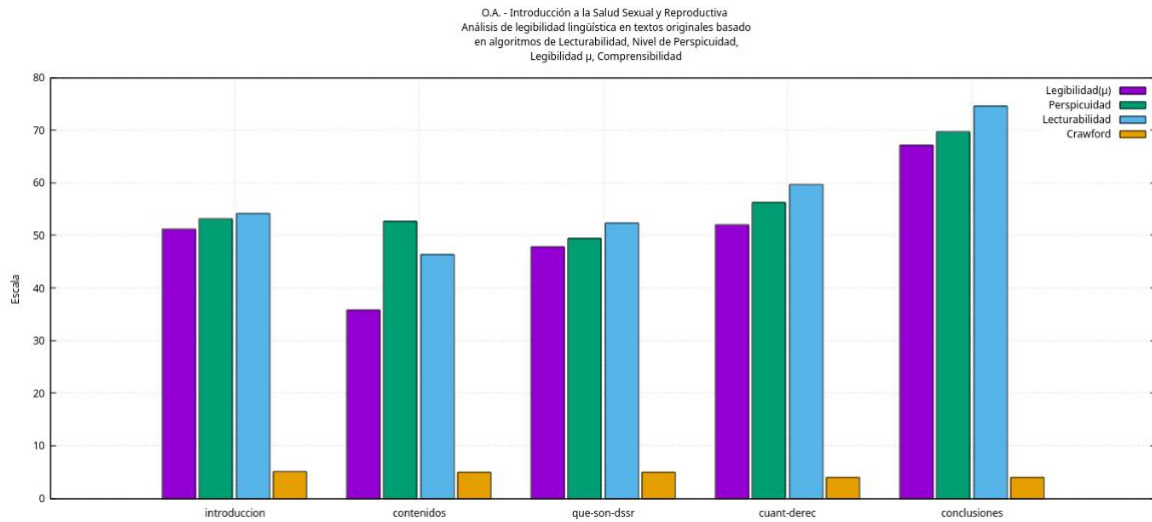


Figura 6.62 Análisis de legibilidad lingüística de textos procesados

Experimentación 19 - O.A. Traumatismos: En la figura 6.61 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.62 se evidencia el índice de valores postimplementación sobre los textos extraídos.

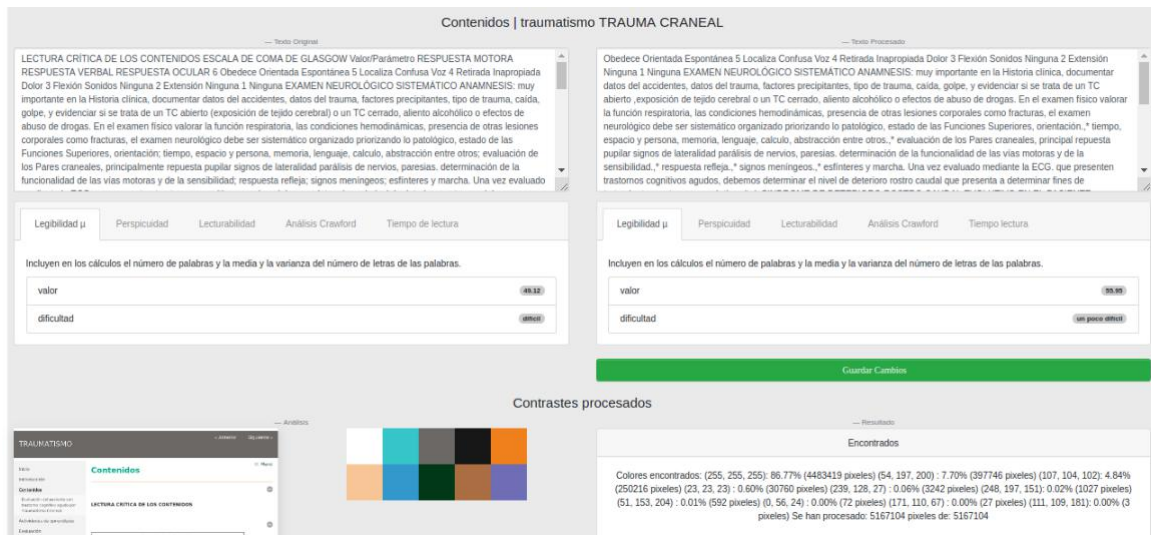


Figura 6.63 O.A. Traumatismos

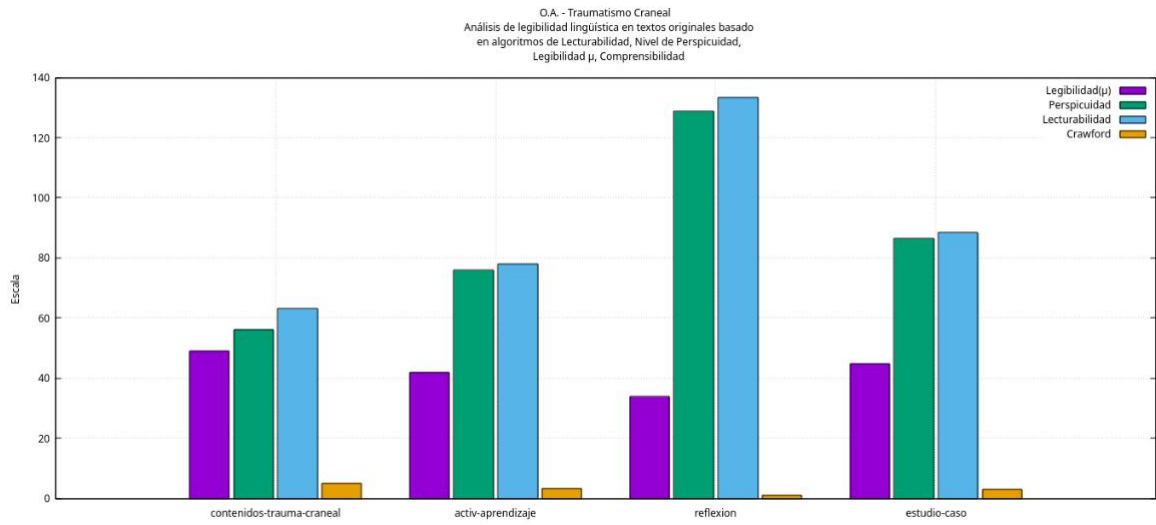


Figura 6.64 Análisis de legibilidad lingüística de textos originales

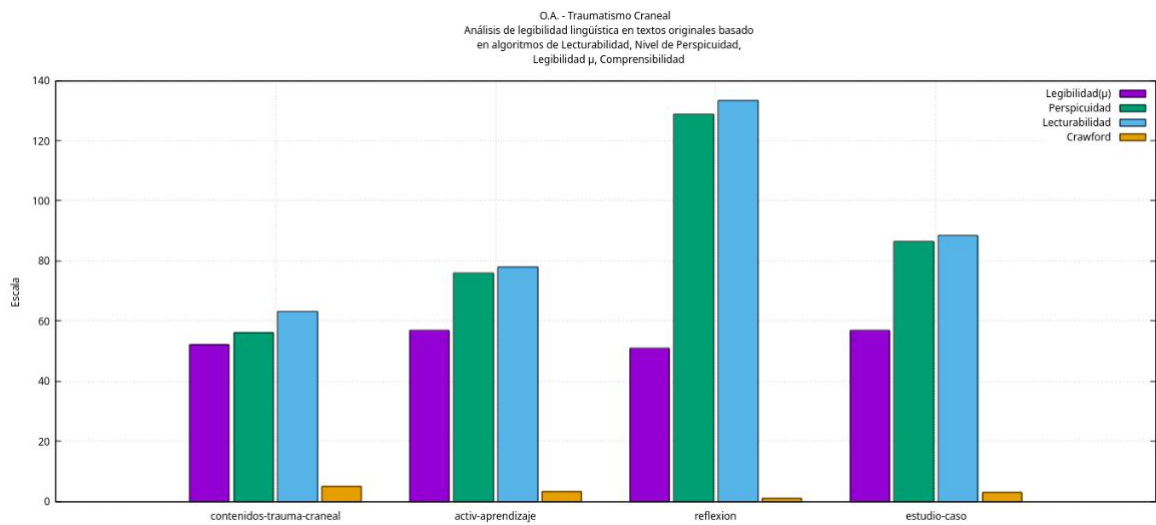


Figura 6.65 Análisis de legibilidad lingüística de textos procesados

Experimentación 20 - O.A. Procesador: En la figura 6.61 se evidencia el análisis de los textos originales extraídos del O.A. Y en la figura 6.62 se evidencia el índice de valores postimplementación sobre los textos extraídos.

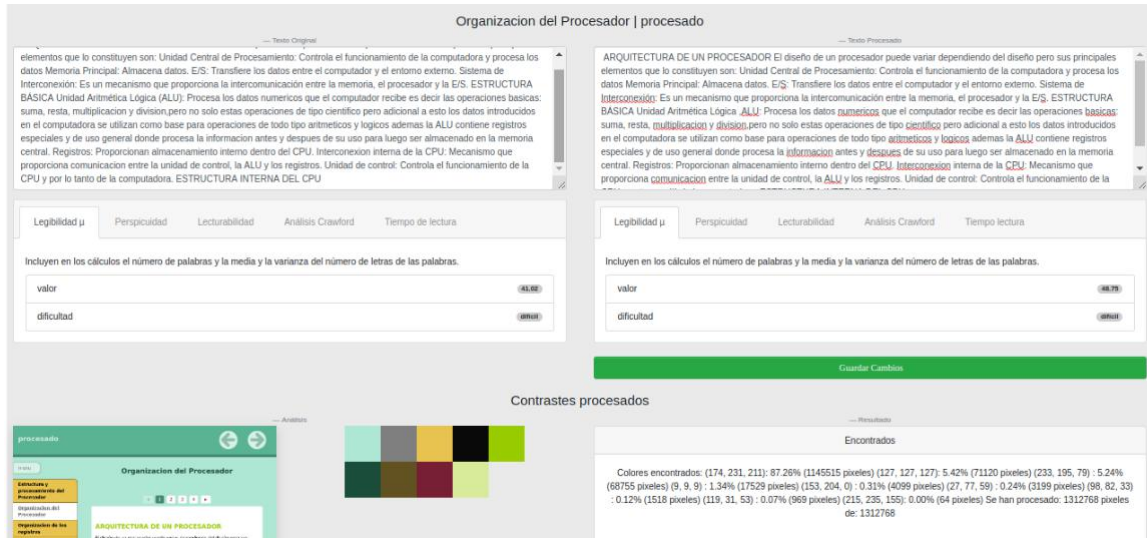


Figura 6.66 O.A. Procesador

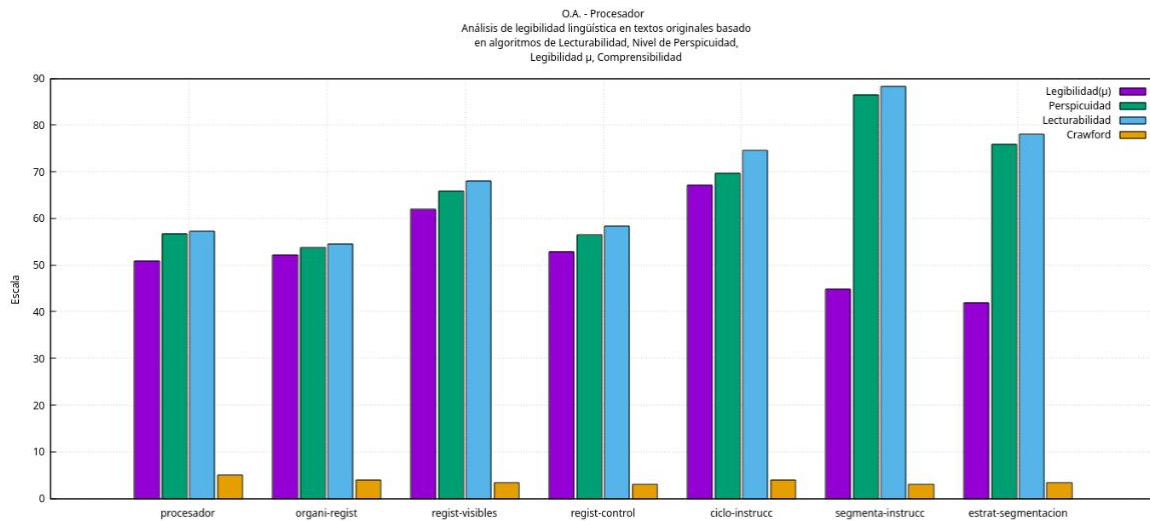


Figura 6.67 Análisis de legibilidad lingüística de textos originales

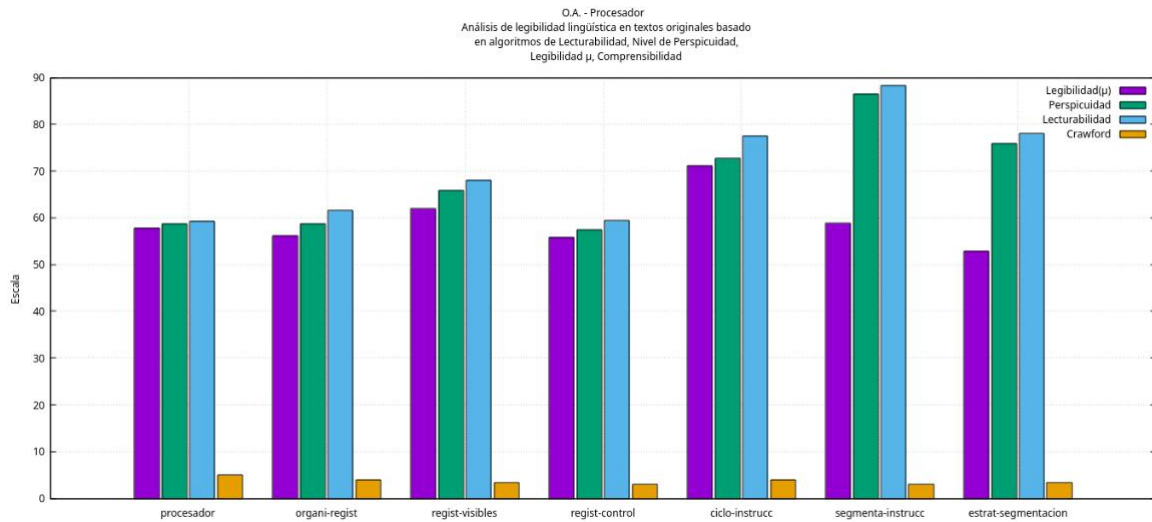


Figura 6.68 Análisis de legibilidad lingüística de textos procesados

Dentro de esta fase de experimentación se define un nuevo perfil de discapacidad para un nuevo análisis, en este caso Discapacidad Auditiva, al tener la selección de dicho perfil obtenemos como resultado las configuraciones por el sistema experto presentadas en la Figura 6.69:

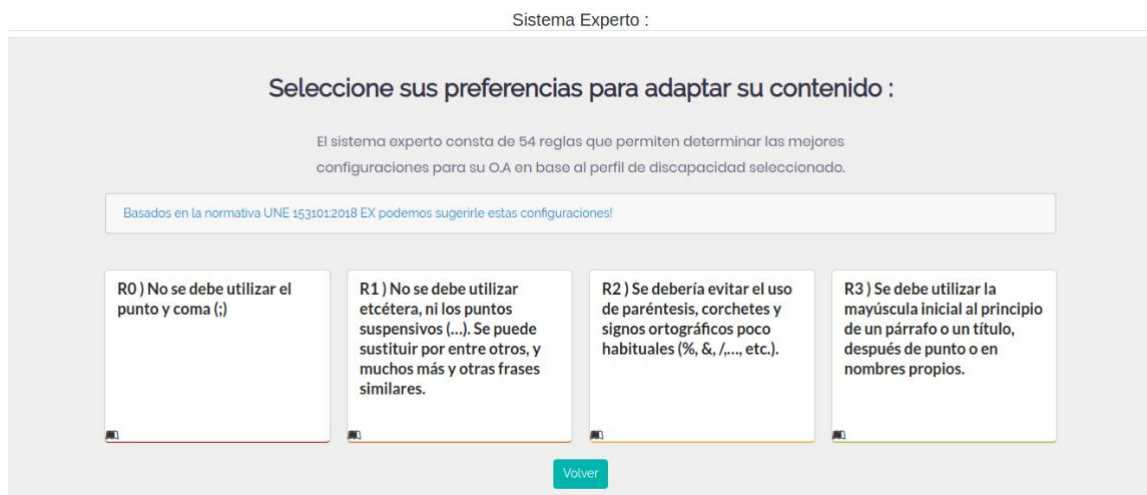


Figura 6.69 Resultado de configuraciones para perfil con discapacidad auditiva

Obteniendo como resultado los siguientes análisis de lecturabilidad para el perfil determinado, a partir de la selección de 10 O.A. aleatorios del R.O.A. generado:

Experimentación 21 - O.A. Algoritmos: En la figura 6.70 se evidencia el índice de valores postimplementación sobre los textos extraídos.

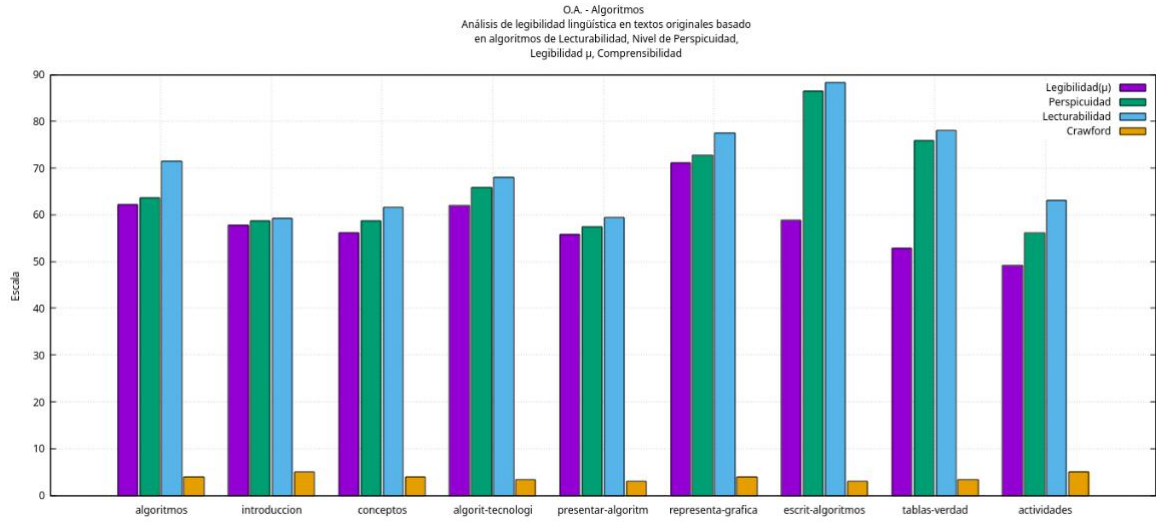


Figura 6.70 Análisis de legibilidad lingüística de textos procesados

Experimentación 22 - O.A. Cromosomas [25]: En la figura 6.71 se evidencia el índice de valores postimplementación sobre los textos extraídos.

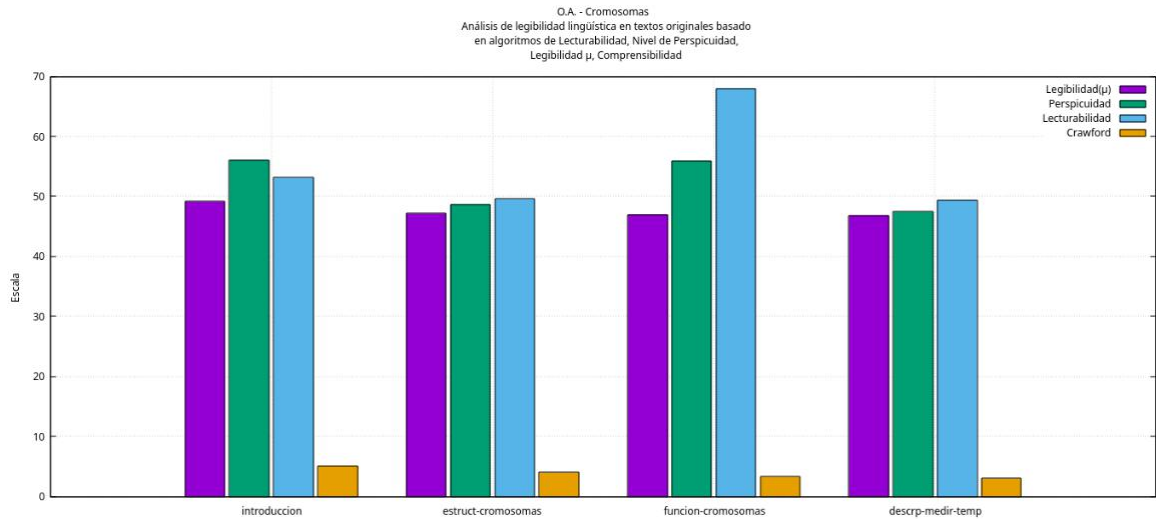


Figura 6.71 Análisis de legibilidad lingüística de textos procesados

Experimentación 23 - O.A. Célula eucariota [25]:

En la figura 6.72 se evidencia el índice de valores postimplementación sobre los textos extraídos.

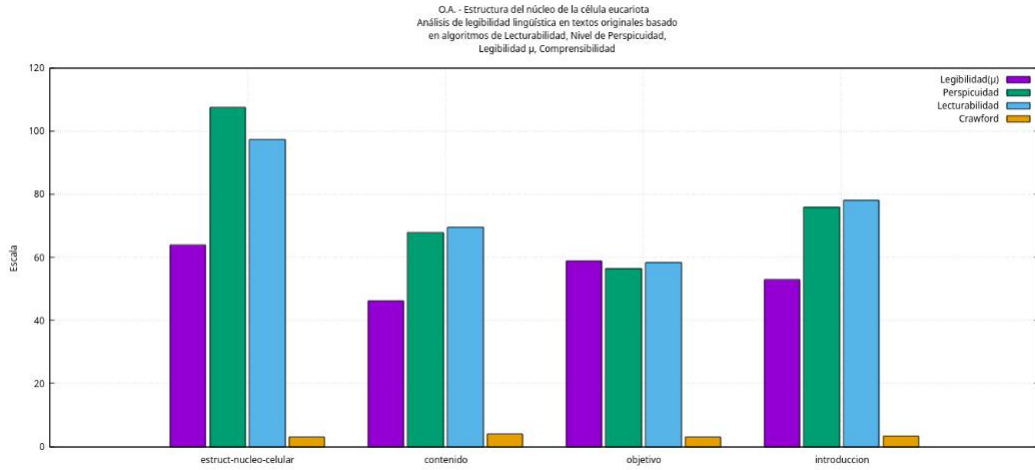


Figura 6.72 Análisis de legibilidad lingüística de textos procesados

Experimentación 24 - O.A. Organización y Arquitectura del Computador:

En la figura 6.73 se evidencia el índice de valores postimplementación sobre los textos extraídos.

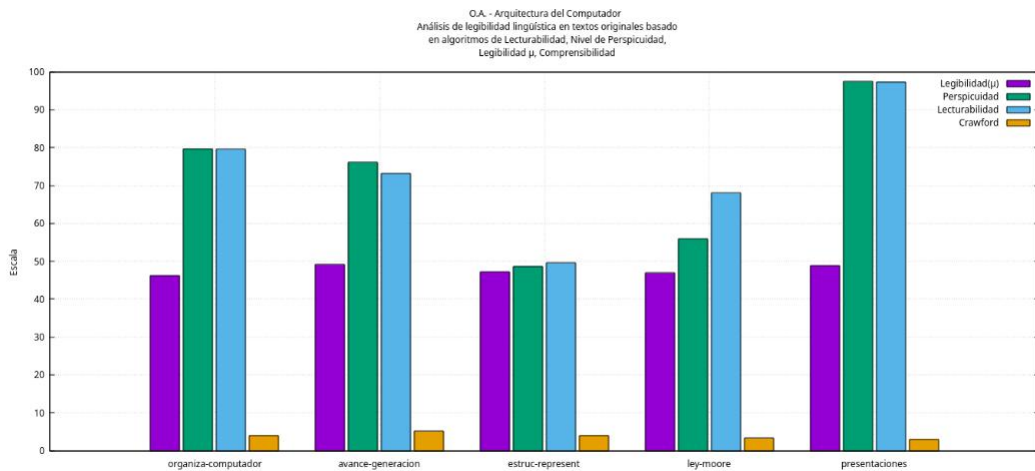


Figura 6.73 Análisis de legibilidad lingüística de textos procesados

Experimentación 25 - O.A. Traumatismos:

En la figura 6.74 se evidencia el índice de valores postimplementación sobre los textos extraídos.

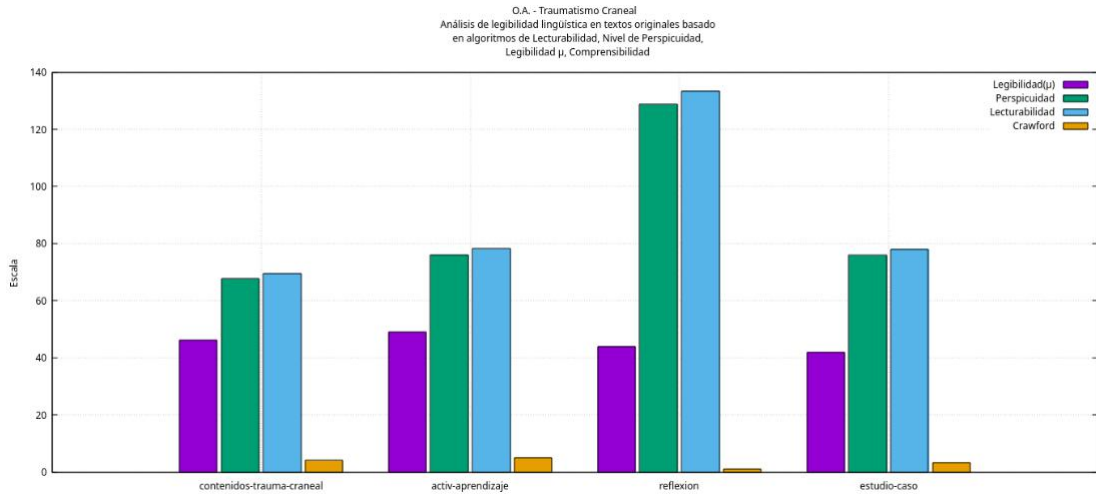


Figura 6.74 Análisis de legibilidad lingüística de textos procesados

Experimentación 26 - O.A. Sistema solar:

En la figura 6.75 se evidencia el índice de valores postimplementación sobre los textos extraídos.

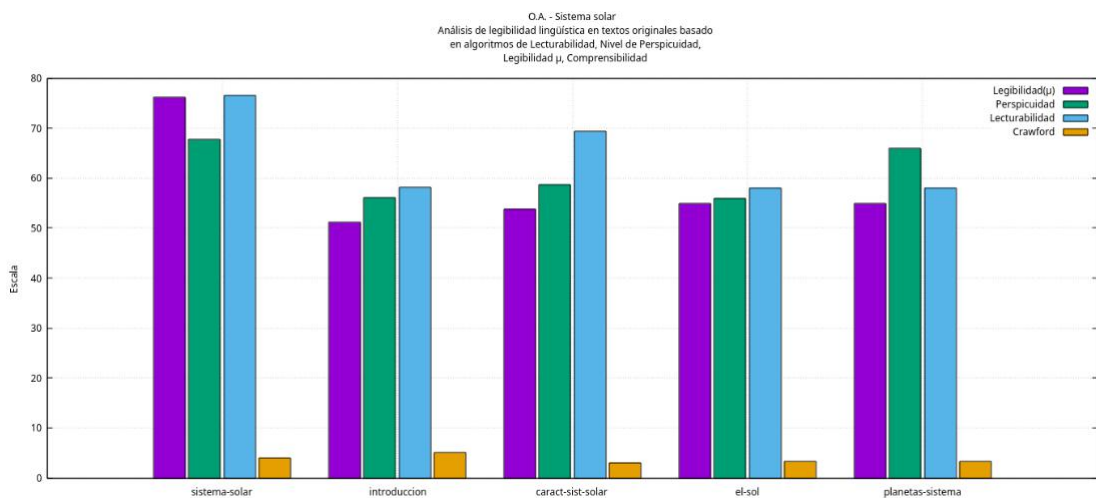


Figura 6.75 Análisis de legibilidad lingüística de textos procesados

Experimentación 27 - O.A. Introducción Salud sexual y reproductiva:

En la figura 6.76 se evidencia el índice de valores postimplementación sobre los textos extraídos.

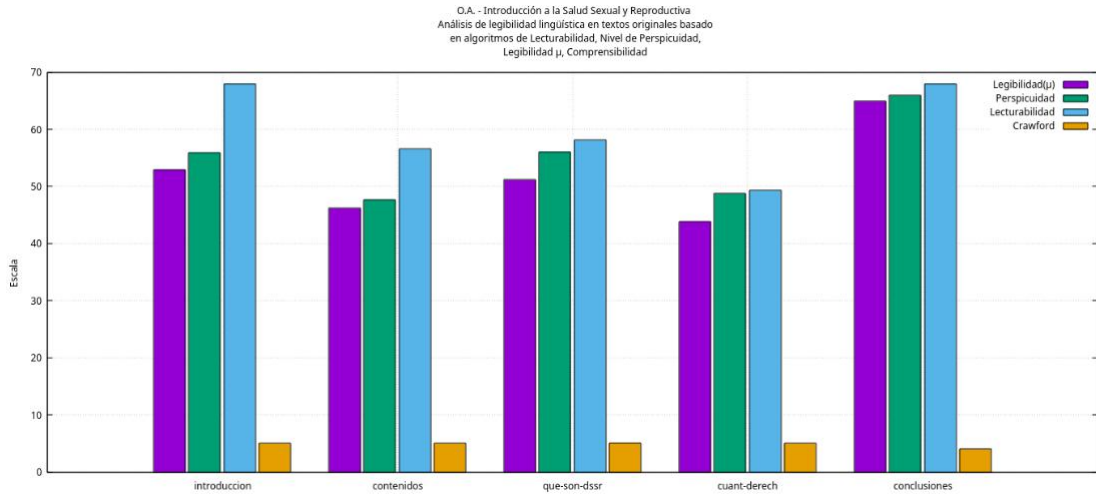


Figura 6.76 Análisis de legibilidad lingüística de textos procesados

Experimentación 28 - O.A. Óxidos:

En la figura 6.77 se evidencia el índice de valores postimplementación sobre los textos extraídos.

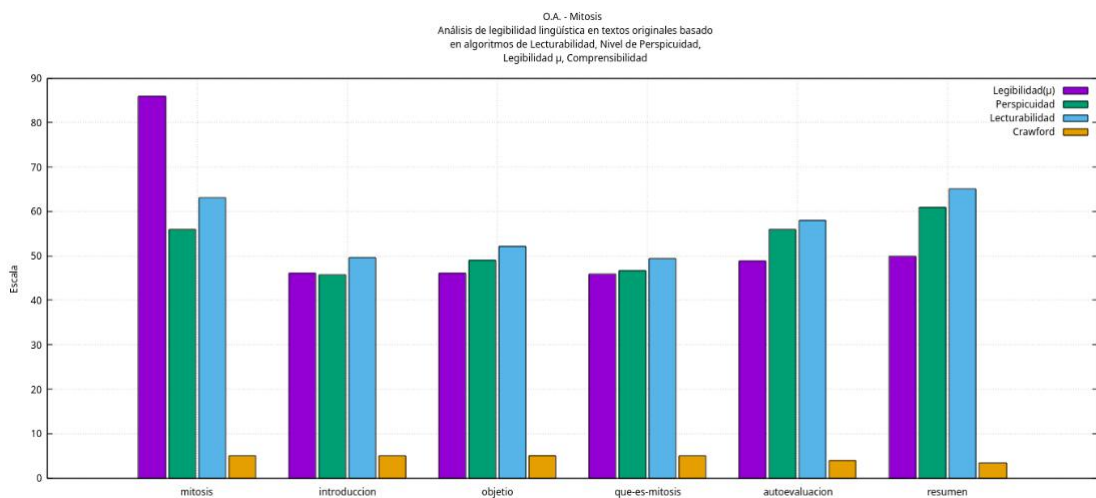


Figura 6.77 Análisis de legibilidad lingüística de textos procesados

Experimentación 29 - O.A. Mitosis:

En la figura 6.78 se evidencia el índice de valores postimplementación sobre los textos extraídos.

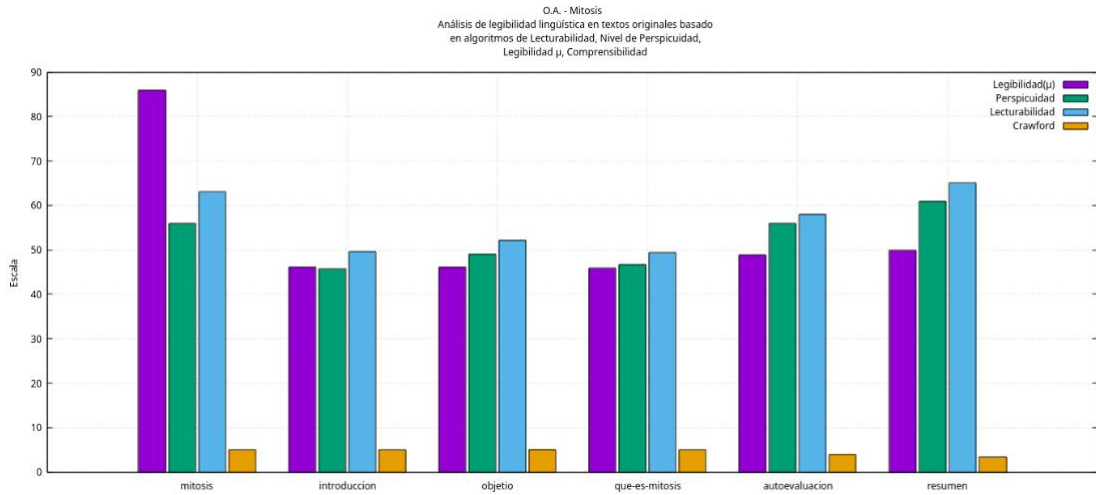


Figura 6.78 Análisis de legibilidad lingüística de textos procesados

Experimentación 30 - O.A. Técnicas de medición de la temperatura corporal:

En la figura 6.79 se evidencia el índice de valores postimplementación sobre los textos extraídos.

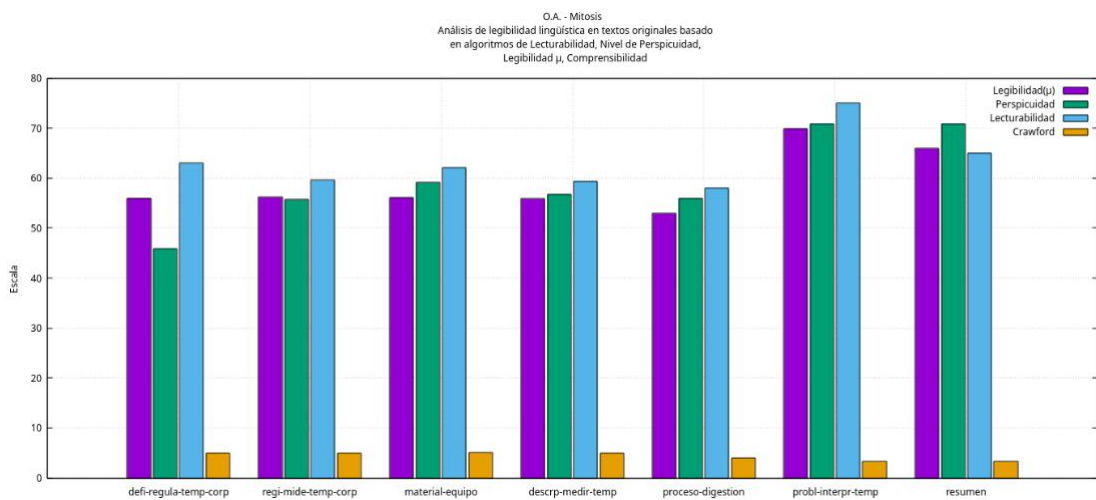


Figura 6.79 Análisis de legibilidad lingüística de textos procesados

Para síntesis de esta experimentación se realizó un análisis de 166 componentes html correspondientes a cada O.A. albergado por el docente, dicha experimentación se refleja en bloques, el primero con un análisis de legibilidad, perspicuidad, lecturabilidad y análisis crawford sobre los textos originales resultantes del módulo de extracción de contenidos, y el segundo análisis tomando de referencia dichos algoritmos sobre los textos resultantes de aplicar las configuraciones dadas por el S.E, al realizar una comparativa entre dichos resultados se refleja un incremento en los índices definido en un rango del 3 y 6 % para cada algoritmo post implementación del sistema inteligente.

Conclusiones

Por medio de esta herramienta se presenta una propuesta que permite a los docentes y educadores especiales agilizar sus procesos en la adaptación de contenidos de un material educativo en formato de O.A. a diferentes perfiles de necesidad o variabilidad del aprendizaje. Mediante esta herramienta se generan contenidos educativos personalizados que son potenciadores en procesos educativos basados en entornos virtuales de aprendizaje.

Gracias al empleo de técnicas inteligentes como un sistema experto basado en reglas, el procesamiento de lenguaje natural y procesamiento de imágenes, la automatización de pautas y normativas para la adaptabilidad de O.A., permiten generar estrategias que potencian la creación de nuevas metodologías educativas. Como se evidencia en este documento generar un proceso de entrenamiento en modelos de lenguaje es llamativo para mejorar la identificación de componentes textuales (entidades, lugares, nombres, entre otros) propios de cada lenguaje, que, en combinación con el uso de expresiones regulares, la manipulación y procesamiento profundo de textos, permite lograr dicha adaptación a escenarios y necesidades determinadas, brindando de esta manera la capacidad de que el sistema pueda adoptar nuevos perfiles de discapacidad para los cuales adaptar los contenidos educativos.

Se evidencia que con el empleo de dichas técnicas los índices como legibilidad y lectura- bilidad para cada perfil de necesidades incrementan en relación a textos sin adaptar. Cabe señalar que es indispensable contar con experto que apoyen la decisión de selección para

determinar cuales de las pautas y configuraciones a automatizar favorecen e intervienen para un perfil de discapacidad determinado.

El desarrollo de este proyecto refleja un proceso complejo y necesario de adaptación de componentes educativos a diferentes perfiles de discapacidad, siendo este el objetivo de la accesibilidad para todos, generando un esfuerzo único que favorezca a la mayoría de grupos históricamente excluidos teniendo en cuenta las exigencias de cada uno de ellos.

La esencia de esta propuesta es que los educadores desarrollen un estado de conciencia sobre las diferencias que existen en abstraer un conocimiento para cada estudiantes y los procesos educativos que acarrea dichas diferencias, se demuestra además que las metodo-logías tradicionales útiles pero no imprescindibles particularmente en el sistema educativo latinoamericano se deben adaptar para promover oportunidades de aprendizaje equitativos y de calidad.

Recomendaciones

Es importante resaltar algunos procesos a tener en cuenta tanto en el área de desarrollo como de implementación.

El docente debe estar familiarizado con la metodología Dicrevoa 2.0, la cual expone una estructura para el desarrollo de O.A que es sobre la cual se basa esta propuesta; dicha estructura debe ser reflejada en el orden del spine para un O.A, siendo este una Introducción, Objetivos, Contenidos, Autoevaluación y una Bibliografía.

Para que el sistema tenga un buen rendimiento dentro del O.A se debe identificar un porcentaje lógico y considerable de textos, esta recomendación se da al identificar escenarios en los cuales se puede dar que un O.A este constituido solo por imágenes y vídeos.

Los productos resultantes con esta propuesta se los pueden potenciar en combinación con software o herramientas que pueden ser TTS (text to speech), Lectores de pantalla, VirtualKeyboard, NVDA's, Mouse Joystick, entre otros componentes que permiten una mejor adaptación para perfiles de discapacidad determinados.

Al darse la posibilidad de replicar este trabajo se debe tener en consideración que con modelos de lenguaje entrenados así como el aumento de reglas en el sistema experto y una base de hechos de un mayor volumen se pueden abarcar mayores perfiles de discapacidad así como potenciar la herramienta a nuevos escenarios como el laboral.

Trabajo futuro

Cabe mencionar que el desarrollo de esta propuesta se basa en una normativa UNE:153101 EX (experimental), esto quiere decir que se deben generar procesos de mejoras y adaptaciones sobre esta normativa brindando la oportunidad de implementar nuevas técnicas y algoritmos que cumplan con las expectativas de varios perfiles de estudiantes.

Entre estas técnicas podemos mencionar la incorporación de herramientas de anotación impulsada y de aprendizaje activo para el reconocimiento de entidades, detección de intenciones o clasificación de imágenes, mejorando el entrenamiento y modelamiento de estructuras de lenguaje de una manera más rápida y sin consumir muchos recursos, ya sean estas herramientas libres o de pago como es el caso de prodigy.

Se propone desarrollar un proceso más elaborado para el módulo de análisis de contrastes automatizados, brindando al docente un reporte sobre las secciones puntuales sobre las cuales se debe realizar una mejora para garantizar una buena implementación de cromáticas en personas con un perfil de discapacidad visual, intelectual o cognitiva.

Bibliografía

- [1] Baquedano, M. M. (2006). Legibilidad y variabilidad de los textos. pages 5–6.
- [2] clips (2018). About clips.
- [3] Corporación Ecuatoriana para el Desarrollo de la Investigación y la Academia (CEDIA) (2019). Repositorio de objetos de aprendizaje.
- [4] Crawford, A. (1985). Fórmula y gráfico para determinar la comprensibilidad de textos del nivel primario en castellano. *Lectura Y Vida*, 4:18–24.
- [5] Faces, J. (2019). Javasever faces.org.
- [6] Fernández Huerta, J. (1959). Medidas sencillas de lecturabilidad. *Consigna*, 214:29–32.
- [7] Hedley, J. et al. (2017). 'jsoup: Java html parser', 2015. Website (<https://jsoup.org/>).
- [8] Ingavélez-Guerra, P., Cuzco-Calle, I., Calle-López, D., Oyola-Flores, C., Yambay-Aulla, I., Robles-Bykbaev, V., and Hilera, J. R. (2017). An intelligent system to automatically generate video-summaries for accessible learning objects for people with hearing loss. In *International Conference on Applied Human Factors and Ergonomics*, pages 113–122. Springer.
- [9] Ingavélez-Guerra, P., Robles-Bykbaev, V., Otón, S., Vera-Rea, P., Galán-Men, J., Ulloa-Amaya, M., and Hilera, J. (2018a). A proposal based on knowledge modeling and ontologies to support the accessibility evaluation process of learning objects. In *2018 Congreso Argentino de Ciencias de la Informática y Desarrollos de Investigación (CACIDI)*, pages 1–5. IEEE.
- [10] Ingavélez-Guerra, P., Ulloa-Amaya, M., Vera-Rea, A., Galán-Mena, J., Robles-Bykbaev, V., Hilera, J., and Otón, S. (2018b). An ontological network to identify accessibility metadata in learning objects: an approach based on web content accessibility guidelines, schemas, and disabilities analysis. In *2018 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*, pages 1–6. IEEE.
- [11] Instituto Nacional de Evaluación Educativa (2018). Educación en Ecuador resultados de pisa para el desarrollo.
- [12] Larreátegui Ortega, P. A. (2016). Derecho a una educación inclusiva para estudiantes discapacitados según la aplicación del reglamento de la loei. B.S. thesis, Quito: UCE.

- [13] Leonard, A. and Alin, C. (2015). *Mastering OmniFaces*. Glasnevin Publishing.
- [14] López-Nores, M., García-Duque, J., Pazos-Arias, J., Arévalo-Lucero, D., Ingavélez-Guerra, P., and Robles-Bykbaev, V. (2016). Una propuesta de un ecosistema basado en herramientas tic inteligentes para apoyar el diagnóstico y la intervención de pacientes con trastornos de la comunicación. *Ingenius*, 1(14):44–52.
- [15] Maldonado, J. (2014). *Metodología para la creación de material educativo digital*.
- [16] Maldonado, J. J., Bermeo, J. L., and Mejía, M. (2015). Dicrevoa: A proposal for the design, creation and evaluation of learning objects. In *2015 Latin American Computing Conference (CLEI)*, pages 1–11. IEEE.
- [17] Montoto, O. C. (2019). *Lectura fácil. pautas y recomendaciones*. une 153101:2018 ex.
- [18] Pazos, F. S. (1993). *Sistemas predictivos de legibilidad del mensaje escrito: fórmula de perspicuidad*. Universidad Complutense de Madrid, Servicio de Reprografía.
- [19] Python (2019). *re — regular expression operations*.
- [20] Rossum, G. (1995). *Python tutorial*.
- [21] spaCy (2019). *spacy 101: Everything you need to know*.
- [22] Team, O. D. (2017). *Opencv*. On line]. Available at: <https://opencv.org/>. [Último acceso: 15 06 2018].
- [23] UNE Normalización española (2018). *Une 153101:2018 ex lectura fácil. pautas y recomendaciones para la elaboración de documentos*.
- [24] UNESCO (2016). *Education for people and planet: creating sustainable futures for all, global education monitoring report*.
- [25] Universidad Popular del Cesar (2019). *Blog sobre objetos de aprendizaje*.
- [26] Zachary Schuessler (2016). *Delta e 101*.

Apéndice A

Presentación de propuesta



Figura A.1 Presentación de propuesta a miembros del Consejo Nacional para la Igualdad de Discapacidades (Conadis), 2019

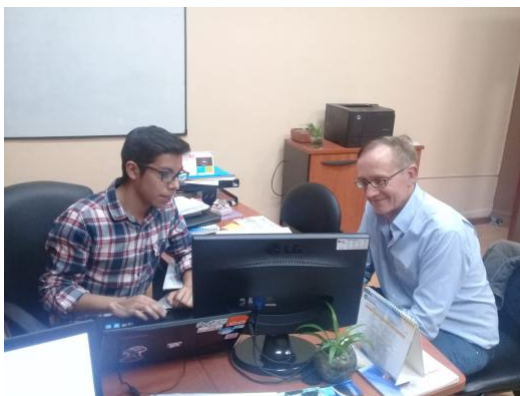


Figura A.2 Presentación de propuesta a experto español de discapacidades, 2019