

**ESCUELA POLITÉCNICA
SUPERIOR DE CÓRDOBA**
Universidad de Córdoba



ANTEPROYECTO FIN DE GRADO

Grado en Ingeniería Informática

**OrdClass: aplicación web para el análisis de
problemas de clasificación ordinal**

**Autor: Jesús Bueno Ruiz
Director: David Guijo Rubio**

Octubre, 2021



UNIVERSIDAD DE CÓRDOBA

ÍNDICE DE CONTENIDOS

1. Datos del proyecto	1
2. Introducción	2
3. Objetivos	5
4. Antecedentes	7
4.1. Información sobre clasificación ordinal	7
4.1.1. <i>Ordinal Regression en Wikipedia</i>	7
4.1.2. <i>Ordinal Regression Methods: Survey and Experimental Study</i>	8
4.2. Repositorios/librerías para la clasificación ordinal	9
4.2.1. <i>ORCA</i>	9
4.2.2. <i>ORCA-python</i>	9
4.2.3. <i>mord: Regresión Ordinal en Python</i>	10
4.2.4. <i>Papers with code</i>	10
4.3. Lenguajes de programación utilizados	11
4.3.1. <i>MATLAB</i>	11
4.3.2. <i>Python</i>	12
5. Fases de desarrollo	13
5.1. Definición y estudio del problema	13
5.2. Análisis de requisitos	14
5.3. Diseño	14
5.4. Implementación	15
5.5. Pruebas	16
5.6. Documentación	16
6. Recursos	17
6.1. Recursos humanos	17
6.2. Recursos materiales	17
6.2.1. <i>Recursos software</i>	17
6.2.2. <i>Recursos hardware</i>	17
7. Cronograma	18
8. Bibliografía	19

1. Datos del proyecto

Título: OrdClass: aplicación web para el análisis de problemas de clasificación ordinal.

Autor:

- **Nombre:** Jesús Bueno Ruiz
- **Email:** i82buruj@uco.es
- **Titulación:** Grado en Ingeniería Informática

Director: David Guijo Rubio

2. Introducción

Los problemas de clasificación ordinal [1] (o regresión ordinal) son muy frecuentes en diversas áreas de investigación de mucha actualidad y tienen un alto interés en muchas aplicaciones del mundo real, como en la medicina, el cálculo de estimación de edad, modelo econométrico, reconocimiento facial, clasificación de imágenes, etc.

La clasificación ordinal es un tipo de análisis que se utiliza para predecir una variable ordinal, en este caso la variable dependiente de un determinado problema. Dicha predicción se realizará en función de un conjunto de variables independientes o de entrada. En términos matemáticos, los problemas de clasificación ordinal consisten en predecir un valor y sobre un vector de entrada x , donde $x \in X \subseteq \mathbb{R}^k$ y $y \in Y = \{C1, C2, \dots, Cq\}$ donde x es un espacio k -dimensional e y es una etiqueta perteneciente a Q diferentes niveles. Además, existe una relación de orden entre las clases de Y , de forma que se cumple la restricción $C1 < C2 < \dots < Cq$, donde $<$ representa la relación de orden.

La clasificación ordinal puede realizarse aplicando diferentes métodos y algoritmos. Cada método presenta una serie de ventajas y desventajas respecto a los demás, dependiendo de las características que estos presenten y de la forma de realizar la clasificación. Por ejemplo, el método *Proportional Odds Model* [2] es una de las mejores opciones si se prioriza el coste computacional. En la Figura 2.1 se muestra una taxonomía de las principales técnicas de clasificación ordinal, extraída del artículo *Ordinal Regression Methods: Survey and Experimental Study* [3].

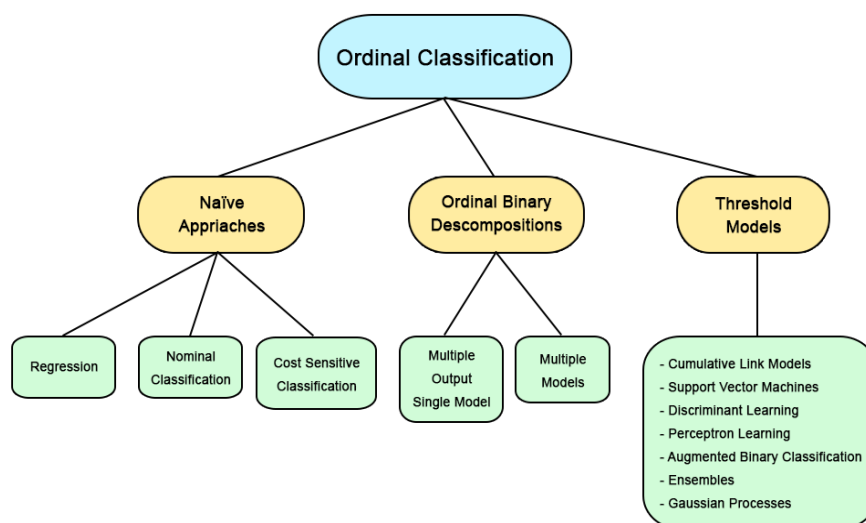


Figura 2.1. Taxonomía de la clasificación ordinal.

Este tipo de problema es cada vez más frecuente, además de que dicho campo ha evolucionado mucho en la última década. Actualmente, no hay ninguna aplicación web que ayude a analizar los conjuntos de datos disponibles para este tipo de problema y muestre los mejores métodos y resultados en función del conjunto de datos que se esté tratando. Por otra parte, existen softwares donde se puede realizar la clasificación ordinal de diferentes conjuntos de datos, pero estos requieren un alto conocimiento de informática y matemáticas. Por ello, una de las principales consideraciones a tener en cuenta es simplificar el procedimiento de la clasificación lo máximo posible, así como proporcionar una herramienta potente para los investigadores de este.

Al ser un problema bastante frecuentado, sobre todo en tecnologías relacionadas con la titulación, y las alternativas que hay requieren de un nivel de conocimiento bastante alto, se propone este proyecto, una aplicación web intuitiva que pueda ser utilizada tanto por expertos en el área (tendrán conjuntos de datos ordinales dispuestos para probar las nuevas metodologías desarrolladas por ellos, tendrán los resultados obtenidos en la actualidad por los métodos del estado del arte, etc), como por principiantes, cuyo principal interés consiste en poder lanzar algunos de los

métodos ya implementados sobre sus conjuntos de datos propios, permitiendo además que puedan subir dichos conjuntos de datos a la aplicación web. De esta forma, a estos últimos usuarios se les facilitaría mucho la tarea al poder incluir documentación suficiente y tutoriales con los que simplificar la tarea de ejecución de métodos posterior.

La aplicación web tendrá funcionalidades como importar datos para su posterior análisis, una lista de los métodos más utilizados para la clasificación ordinal, análisis de resultados, entre otras. Se incluirán conjuntos de datos predefinidos, pero se podrán cargar conjuntos de datos externos si el usuario así lo desea, con la finalidad de incrementar tanto como sea posible el total de conjuntos de datos disponibles para la evaluación de métodos.

Aunque la temática del proyecto guarde relación con la clasificación ordinal y sus diferentes métodos, el proyecto se centrará principalmente en la construcción de un sistema web aplicando todos los métodos ingenieriles aprendidos durante la titulación, y especialmente durante la especialidad en Ingeniería del Software.

Los conjuntos de datos predefinidos y los métodos de clasificación ordinal serán proporcionados por AYRNA [4], por lo que el proyecto se centrará en la construcción y diseño de la aplicación web.

La aplicación web se irá visualizando a través del servidor local durante el desarrollo del proyecto. Los datos necesarios serán extraídos de fuentes externas, y los datos que se necesiten guardar se almacenarán en una base de datos local.

3. Objetivos

El objetivo principal del proyecto es realizar una aplicación web aplicando todos los conceptos de ingeniería de software capacitada para facilitar el análisis de un conjunto de datos a partir de métodos de clasificación ordinal. Además, el proyecto a realizar debe cumplir objetivos como:

- El sistema podrá ser utilizado por cualquier persona. Se tendrán en cuenta aspectos de diseño para lograr una aplicación web que se adapte, en medida de lo posible, a personas con algún tipo de diversidad funcional.
- El sistema deberá ofrecer una interfaz simple e intuitiva, de manera que cualquier usuario, independientemente de los conocimientos informáticos que posea, pueda utilizar la aplicación web.
- La aplicación web va a permitir descargar conjuntos de datos ordinales en formato temporal (series temporales), en formato imagen y en formato estándar.
- El usuario podrá disponer de los resultados para un gran número de métodos de clasificación ordinal sobre los conjuntos de datos mencionados anteriormente.
- El sistema mostrará el código en GitHub [5] de muchos de los métodos de clasificación ordinal (disponibles tanto en python, ORCA-Python [6], como en MATLAB/OCTAVE [7], ORCA [8]).

- La aplicación web debe permitir obtener ficheros de configuración a partir de la inserción de los datos en una de las partes de la aplicación web. Es decir, pasar de un lenguaje natural a lenguaje de programación.
- El sistema debe mejorar los servicios que ofrecen otras alternativas, mejorando la comodidad y la rapidez para realizar operaciones con la clasificación ordinal.
- Se debe permitir a los usuarios interactuar con la aplicación para subir conjuntos de datos y métodos, a ser posibles integrados en ORCA o ORCA-python.
- Se debe fomentar el aprendizaje y el estudio de este campo de la estadística, de manera que un usuario pueda aprender a interpretar los resultados obtenidos y defenderse con los conceptos básicos.
- Además del aprendizaje, se pretende fomentar la investigación sobre la clasificación ordinal y los resultados que se obtienen a partir de esta.
- Apostar por un buen uso de los datos recogidos, sin infringir derechos de autor y haciendo siempre buen uso de ellos.
- Lograr la atención del usuario a partir de tableros con últimas noticias, novedades, etc.

4. Antecedentes

Actualmente no existe ninguna alternativa precisa a la aplicación web que pretendemos desarrollar. Tan solo existen nodos de información independientes entre los cuales se incluye información de clasificación ordinal, en otros incluyen métodos, otros contienen bases de datos, etc

El objetivo es solventar esta desconexión existente entre todas estas partes y generar una herramienta muy potente en el mundo de la clasificación ordinal, facilitando el trabajo tanto a los expertos en el área como a los principiantes.

Se ha elaborado una lista con las diferentes alternativas relacionadas con problemas de clasificación ordinal, con el fin de ver las principales ventajas y desventajas de las fuentes de información y alternativas de mercado actuales.

Los antecedentes analizados se pueden dividir en tres secciones: en primer lugar, se encuentran aquellos de los que se ha extraído información para el estudio y posterior desarrollo del proyecto, en segundo lugar, los repositorios y librerías necesarias, por último, se encuentran los lenguajes que se utilizarán para la implementación de clasificación ordinal.

4.1. Información sobre clasificación ordinal

4.1.1. *Ordinal Regression* en Wikipedia

Ordinal Regression se trata de un artículo extraído de Wikipedia [9]. En él, se define el concepto de clasificación ordinal, exponiendo además ejemplos sencillos para su comprensión. También trata modelos lineales para la clasificación ordinal, exponiendo conceptos más avanzados para un público más experto.

Con este artículo se puede conocer los conceptos básicos y algunos avanzados de la clasificación ordinal, pero falta información como la taxonomía de la clasificación ordinal.

4.1.2. Ordinal Regression Methods: Survey and Experimental Study

Ordinal Regression Methods: Survey and Experimental Study [3] es un artículo científico donde se tratan los problemas de clasificación ordinal, aportando una definición sobre el problema (ejemplos del mundo real y notación matemática) y explicando los principales métodos de clasificación.

Es un artículo de gran utilidad ya que gran parte de la información acerca de este proyecto ha sido extraída de él, además de que explica de una manera clara y concisa los problemas de clasificación ordinal y su relación con diferentes ciencias. Sobre todo, destaca por su extensa explicación acerca de la taxonomía de la clasificación ordinal, que ayuda bastante a la comprensión del problema. Además, aporta una conclusión final sobre los métodos de clasificación ordinal expuestos, comparando las ventajas y desventajas de cada método, y para que se debe utilizar cada uno.

4.2. Repositorios/librerías para la clasificación ordinal

4.2.1. ORCA

ORCA (Ordinal Regression and Classification Algorithms) [8] es un framework para MATLAB desarrollado por AYRNA. Este framework incluye un amplio conjunto de métodos de clasificación ordinal asociados al artículo antes mencionado (*Ordinal Regression Methods: Survey and Experimental Study*).

ORCA se desarrolló con el objetivo de poder realizar experimentos con el fin de facilitar la comparación entre algoritmos y conjuntos de datos. Se trata de un software libre, por lo que su código es público y se pueden incluir fácilmente nuevos algoritmos en el framework.

Presenta ventajas como que no se trata de un software como tal, sino que es un framework que se puede integrar en MATLAB, y como se ha comentado anteriormente, se puede obtener de forma gratuita y modificar el código si se requiere. Aún así, requiere altos conocimientos de programación.

4.2.2. ORCA-python

ORCA-python [9] está directamente relacionado con ORCA, solo que éste está desarrollado en Python [10]. Está integrado con los módulos scikit-learn [11] y sacred [12].

Se trata de un framework experimental para estudiar el aprendizaje automático a través de clasificación ordinal mediante archivos de configuración sencillos. Al igual que ORCA, está desarrollado por

AYRNA y se puede obtener de manera gratuita ya que es un software libre.

Presenta ventajas y desventajas similares a ORCA, y servirá de gran ayuda ya que a partir de este framework el usuario podrá analizar conjuntos de datos, tarea facilitada mediante la descarga de ficheros de configuración de la aplicación web.

4.2.3. mord: Regresión Ordinal en Python

mord [13] es un paquete de Python que incluye algunos métodos de clasificación ordinal siguiendo la API scikit-learn .

Este paquete proporciona diferentes modelos para solventar problemas de clasificación ordinal. Los modelos los clasifica en: modelos basados en umbrales, modelos basados en regresión y modelos basados en clasificación. Se trata de un software libre, donde su código se puede encontrar en GitHub, pero, al igual que otros antecedentes anteriores, requiere un alto conocimiento de programación.

4.2.4. *Papers with code*

Papers with Code [14] ofrece recursos gratuitos para cualquier usuario. En la web se pueden encontrar documentos, código, conjunto de datos, métodos, etc. Esta página ayuda al estudio e investigación sobre diversos campos, incluida la clasificación ordinal, donde se pueden disponer de artículos interesantes relacionados, como la aplicación de clasificación ordinal para la estimación de edad [15], o para la estimación del rango del índice de masa corporal [16]. Además, todo el mundo

puede contribuir publicando recursos. Su principal desventaja es que es meramente informativa.

4.3. Lenguajes de programación utilizados

4.3.1. MATLAB

MATLAB [7] es un sistema de cómputo numérico que ofrece un entorno integrado con un propio lenguaje de programación. Tiene diversas funcionalidades relacionadas con las matemáticas, como la manipulación de matrices, representación de datos y funciones, implementación de algoritmos, etc. Es un entorno muy popular, usado para la investigación y el desarrollo.

Su principal ventaja es que es un entorno desarrollado específicamente para la manipulación de matrices y el análisis y la representación de datos, además de que está preparado para trabajar con un conjunto de datos de gran tamaño.

MATLAB es un entorno difícil de usar, por lo que necesita un tiempo considerable de estudio, además de que se requiere un alto conocimiento de programación y matemáticas. MATLAB es un software de pago.

4.3.2. Python

Python [10] es un lenguaje de programación multiparadigma ya que soporta la orientación a objetos, programación imperativa y programación funcional. Se trata de un lenguaje interpretado, dinámico y multiplataforma.

Python posee una licencia de código abierto y es uno de los lenguajes de programación más populares y potentes en la actualidad. Es un lenguaje fácil de aprender (en comparación con otros lenguajes populares) y se utiliza en diversos campos de la informática y las matemáticas.

Su principal ventaja, es que como se ha dicho anteriormente, es fácil de aprender y ofrece una sintaxis bastante agradable y sencilla. Además, incluye multitud de frameworks orientados a las matemáticas y problemas de clasificación ordinal.

5. Fases de desarrollo

5.1. Definición y estudio del problema

En primer lugar se deberá definir el propósito de la aplicación web, realizar un estudio sobre el contexto y analizar las herramientas que pueden servir para obtener la máxima calidad posible del proyecto.

Una aplicación web puede dividirse en dos partes, que deben estudiarse para elegir los lenguajes de programación y las herramientas apropiadas:

- **Frontend.** Es la parte donde el usuario interactúa directamente con la aplicación web [17]. Debe cumplir unos estándares básicos de usabilidad y estética.
- **Backend.** Se encarga de que toda la lógica de la aplicación web funcione adecuadamente [17]. No es directamente accesible por el usuario. Debe cumplir unos patrones de diseño para que la aplicación web sea rápida, segura, y fácil de adaptar a futuras versiones.

Otro aspecto a considerar es el estudio sobre el entorno de desarrollo que se va a utilizar. Para el proyecto se utilizará *Visual Studio Code (v1.60.1)* [18].

Por último, se deben estudiar las herramientas que se van a utilizar para cada una de las siguientes fases, tales como herramientas de gestión de requisitos, de modelado, de prototipado, etc.

5.2. Análisis de requisitos

Para esta fase, se aplicarán todas las actividades de la Ingeniería de Requisitos (extracción, análisis, especificación y validación). A través de estas actividades, se detallarán todos los requerimientos que han de cumplirse para que la aplicación web pueda utilizarse y tenga una funcionalidad óptima y atractiva para el usuario. Se definirán varios tipos de requisitos: funcionales, no funcionales y de información [19].

Se realizará una gestión de requisitos exhaustiva empleando técnicas vistas durante la titulación para ir comprobando durante el proyecto que estos se van cumpliendo. Para el modelado de los requisitos, se utilizarán lenguajes de modelado como UML (*Unified Modeling Language*) [20] y UWE (*UML-Based Web Engineering*) [21].

5.3. Diseño

El diseño de la aplicación web debe cumplir con los requisitos de usuario y del sistema. Además, se debe plantear un diseño limpio y eficaz, que atraiga al usuario y se adapte a todos los usuarios posibles.

Para el prototipado y diseño de interfaces, se utilizará la herramienta gratuita *Pencil* [22].

5.4. Implementación

Para la implementación, a partir del estudio mencionado en el apartado 5.1, se utilizarán los siguientes lenguajes de programación y herramientas:

- **Frontend:** para el desarrollo del *frontend*, se emplearán los siguientes lenguajes y herramientas:
 - HTML5 [23] para dar estructura y significado a la web.
 - CSS3 [24] para aplicar estilos.
 - JavaScript [25] para crear funciones dinámicas.
 - Bootstrap [26], *framework* de código abierto que facilita el desarrollo web.
- **Backend:** para el desarrollo del *backend* se utilizará el lenguaje Java, muy usado para aplicaciones web y que presenta gran robustez y seguridad. El *backend*, además, está formado por la base de datos.
- **Base de datos:** para la gestión de la base de datos, se utilizará PHPMyAdmin [27], centrado en el lenguaje MySQL. Además, viene integrado en el programa XAMPP [28], el cual se utilizará para desarrollar el proyecto.
- **Servidor web local:** para utilizar la aplicación web de manera local se utilizará un servidor Apache integrado en XAMPP.
- **Gestión de versiones:** para un desarrollo eficiente, se utilizará la herramienta GitHub, que servirá para almacenar el código del proyecto y gestionar las versiones de este.

5.5. Pruebas

El proyecto debe ser revisado constantemente de manera que cumpla con los requisitos que se especifican en la fase de análisis de requisitos (*apartado 5.2*). Se comprobará que la aplicación web cumpla con los requisitos y se emplearán técnicas para comprobar que la aplicación web pueda usarse de manera óptima y eficaz, cumpliendo así con los principios básicos de la ingeniería web (funcionalidad, fiabilidad, eficiencia, utilidad, actualizabilidad).

5.6. Documentación

Durante el desarrollo del proyecto, se realizará una documentación donde se irá detallando el proyecto web, con el fin de facilitar la comprensión del mismo y de redactar detalladamente los conceptos que involucran al proyecto, la elaboración y el uso.

Durante el desarrollo se aplicarán las principales técnicas de ingeniería vistas durante la titulación, tales como ingeniería de requisitos, modelos de desarrollo software, etc.

6. Recursos

6.1. Recursos humanos

- **Autor:** Jesús Bueno Ruiz. Alumno de Ingeniería Informática de la Universidad de Córdoba.
- **Director:** David Guijo Rubio. Profesor Sustituto Interino del Departamento de Informática y Análisis Numérico y miembro del grupo de investigación AYRNA de la Universidad de Córdoba.

6.2. Recursos materiales

6.2.1. Recursos software

- **Sistema Operativo:** *Windows 10 Home*.
- **Entorno de desarrollo:** *Visual Studio (v1.60.1)*.
- **Entorno de desarrollo web local:** *XAMPP (v 7.3.28)*.

6.2.2. Recursos hardware

Se utilizará un ordenador de mesa con las siguientes características:

- **Procesador:** Intel(R) Core(TM) i5-6400 CPU @2.70GHz 2.71 GHz.
- **Memoria RAM:** 8GB DDR3.
- **Gráfica:** Intel(R) HD Graphics 530.

7. Cronograma

El Trabajo Fin de Grado consta de 12 créditos, por lo que le corresponden 300 horas de trabajo. A continuación, en la Figura 7.1, se detalla la relación entre las tareas mencionadas en el apartado anterior y las horas que se le asignan divididas en cuatro meses, que es el tiempo que se estima necesario para realizar el TFG.

Tarea	Mes 1	Mes 2	Mes 3	Mes 4	Total horas por tarea
Definición del problema	5				5
Estudio del problema	10				10
Análisis de requisitos	30				30
Diseño		30	20		50
Implementación		30	30	25	85
Pruebas			10	30	40
Documentación	20	20	20	20	80
Total horas por mes	65	80	80	75	300

Figura 7.1. Planificación de tareas.

8. Bibliografía

- [1] *Regresión ordinal*. https://en.wikipedia.org/wiki/Ordinal_regression
- [2] *Proportional-odds models*.
http://www.stat.uchicago.edu/~pmcc/reports/prop_odds.pdf
- [3] *Ordinal Regression Methods: Survey and Experimental Study*.
P.A. Gutiérrez, M. Pérez-Ortiz, J. Sánchez-Monedero, F. Fernandez-Navarro and C. Hervás-Martínez. "Ordinal regression methods: survey and experimental study", IEEE Transactions on Knowledge and Data Engineering, Vol. 28, January, 2016, pp. 127-146.
<http://dx.doi.org/10.1109/TKDE.2015.2457911>
- [4] *Grupo AYRNA*. <https://www.uco.es/ayrna/>
- [5] *GitHub*. <https://github.com/>
- [6] *ORCA-python*. <https://github.com/ayrna/orca-python>
- [7] *MATLAB*. <https://es.mathworks.com/products/matlab.html>
- [8] *ORCA*. <https://github.com/ayrna/orca>
- [9] *Ordinal Regression Wikipedia*. https://en.wikipedia.org/wiki/Ordinal_regression
- [10] *Python*. <https://www.python.org/>
- [11] *scikit-learn*. <https://scikit-learn.org/stable/modules/classes.html>

- [12] *sacred*. <https://sacred.readthedocs.io/en/stable/quickstart.html>
- [13] *mord: Ordinal Regression in Python*. <https://pythonhosted.org/mord/>
- [14] *Time Series Classification Papers with Code*.
<https://paperswithcode.com/task/time-series-classification#:~:text=Time%20Series%20Classification%20is%20a,groups%2C%20using%20labeled%20training%20data>.
- [15] *Ordinal Regression With Multiple Output CNN for Age Estimation*.
<https://paperswithcode.com/paper/ordinal-regression-with-multiple-output-cnn>
- [16] *Ordinal Regression using Noisy Pairwise Comparisons for Body Mass Index Range Estimation*.
<https://paperswithcode.com/paper/ordinal-regression-using-noisy-pairwise>
- [17] *Qué es Frontend y Backend*.
<https://platzi.com/blog/que-es-frontend-y-backend/#:~:text=Frontend%20es%20la%20parte%20de,corre%20del%20lado%20del%20servidor>
- [18] *Visual Studio Code*. <https://code.visualstudio.com/>
- [19] *Análisis de requisitos*. <https://boluda.com/tutorial/analisis-de-requisitos/>
- [20] *UML*. https://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado
- [21] *UWE*. <https://uwe.pst.ifi.lmu.de/>
- [22] *Pencil Project*. <https://pencil.evolus.vn/>

- [23] *HTML5*. <https://es.wikipedia.org/wiki/HTML5>

- [24] *Qué es CSS3*.
<https://openwebinars.net/blog/que-es-css3/#:~:text=Definici%C3%B3n%20de%20CSS3&text=CSS%20es%20un%20lenguaje%20de,de%20usuario%20es%20cr%C3%ADtico%20en%20HTML>.

- [25] *JavaScript*. <https://developer.mozilla.org/es/docs/Web/JavaScript>

- [26] *Bootstrap*. <https://getbootstrap.com/>

- [27] *PHPMyAdmin*. <https://www.phpmyadmin.net/>

- [28] *XAMPP*. <https://www.apachefriends.org/es/index.html>