

Aplicativo web para la consolidación de datos y la gestión de la acreditación académica
de los programas UIS.

Hordan Andres Navarro Herrera y Luis Duvan Fernando Pinto Diaz

Trabajo de Grado para Optar al Título de Ingeniero de Sistemas

Director

Carlos Alfonso Mantilla Duarte

Economista - M.Sc. en Estadística Aplicada y Especialista en Estadística

Codirector

Lola Xiomara Bautista Rozo

Ingeniera de Sistemas – PhD en Procesamiento de Imágenes y Señales

Universidad Industrial de Santander

Facultad de Ingenierías Físico-Mecánicas

Escuela de Ingeniería de Sistemas e Infomática

Ingeniería de Sistemas

Bucaramanga

2023

Agradecimientos

Me gustaría aprovechar estas líneas para expresar mi gratitud a todas las personas que nos han brindado ayuda y apoyo durante estos años en especial:

A nuestros padres por confiar y no perder la fe.

A nuestros hermanos por fregarnos la vida.

A nuestros profesores quienes nos guiaron durante este arduo camino hasta aquí.

A nuestros amigos por hacer de este un paso emocionante.

A la universidad, por ser nuestro segundo hogar.

Y a todos ustedes quienes están leyendo por ser parte de esta nuestra última etapa en la universidad.

Tabla de Contenido

	Pag.
Introducción	11
1. Planteamiento y Justificación del Problema	13
2. Objetivos	14
2.1 Objetivo General	14
2.2 Objetivos Específicos	14
3. Propósito del Sistema	15
4. Marco de Referencia	16
4.1 SNIES	16
4.2 Web Scraping	16
4.3 ¿Quiénes para que están usando web scraping con inteligencia artificial?	17
5. Tecnologías a Usar	18
5.1 Python	18
5.2 JavaScript	19
5.3 PHP	20
5.4 HTML5	20
5.5 CSS	21
5.6 Django	21
5.7 Selenium Webdriver	22
5.8 Scrapy	22
5.9 API	23
5.10 Beautiful Soup	23
5.11 Modelo Vista-Controlador	23

5.12	Django Token Authentication.....	24
5.13	Django Rest Framework	25
5.14	Arquitectura De Microservicios.....	25
5.15	JSON.....	26
5.16	Mineria De Datos.....	27
5.17	Inteligencia Artificial	28
5.18	Docker.....	28
5.19	Crontab.....	29
5.20	TAPEX.....	29
5.21	Docker Compose.....	30
5.22	GitHub.....	31
6.	Metodología.....	31
6.1	Metodologia De Desarrollo SCRUM.....	31
6.1.1	Fase I. Investigacion.	34
6.1.2	Fase II. Análisis de Requerimientos.	34
6.1.3	Fase III. Diseño.....	34
6.1.4	Fase IV. Desarrollo.	34
6.1.5	Fase V. Prueba y Despliegue.	35
7.	Modelos del Sistema.....	35
7.1	Descripción de Casos de Uso.....	35
7.1.1	Caso de Uso 1: Hacer Login en la aplicación (Administrador).....	35
7.1.2	Caso deUso 2: El administrador registra o crea las cuentas de los usuarios (Administrador).	
	36

7.1.3	Caso de Uso 3: Hacer Login en la Aplicación (Usuarios).	37
7.1.4	Caso de Uso 4: Realizar una Consulta a una Página (Usuarios).	37
7.1.5	Caso de Uso 5: El administrador edita un perfil de usuario (Administrador).	38
7.1.6	Caso de Uso 6: El administrador elimina un perfil de usuario (Administrador).	39
7.1.7	Caso de Uso 7: Realizar una Grafica a una Página Consultada (Usuarios).	39
7.1.8	Caso de Uso 8: Realizar una Consulta con el asistente de IA (Usuarios, Administrador).	40
8.	Análisis Diseño e Implementación	41
8.1	Análisis e Implementación.	41
8.2	Requisitos Frecuentes de las Interfaces	45
8.3	Requisitos Frecuentes de las Interfaces	46
8.4	Requisitos No Funcionales	46
8.5	Diagramación UML	47
8.5.1	Diagramas de Casos de Uso.	48
8.6	Diseño	52
8.6.1	Manual de Usuario	52
9.	Conclusiones	70
10.	Trabajos Futuros	71
	Referencias Bibliográficas	72

Lista de Tablas

	Pag.
Tabla 1. <i>Tipos de Usuarios del Sistema</i>	44

Lista de Figuras

	Pag.
Figura 1. <i>Diagrama de funcionamiento del modelo de inteligencia artificial Table Pre-training via Learning a Neural {SQL} Executor.</i>	30
Figura 2. <i>El ciclo SCRUM.</i>	33
Figura 3. <i>Diagrama Casos de Uso para el Administrador.</i>	48
Figura 4. <i>Diagrama de Casos de Uso para el Usuario Autenticado.</i>	49
Figura 5. <i>Diagrama de Casos de Uso Registrar Usuarios</i>	50
Figura 6. <i>Diagrama de Casos de Uso Iniciar Sesión.</i>	51
Figura 7. <i>Diagrama de Casos de Uso Consultar, Filtrar y Graficar Información.</i>	51
Figura 8. <i>Diagrama de Casos de Uso Consultar Asistente de IA.</i>	52
Figura 9. <i>Inicio de sesión del administrador y usuarios inscritos en el sistema.</i>	53
Figura 10. <i>Página principal o escritorio del perfil administrador de la aplicación.</i>	54
Figura 11. <i>Página principal del perfil administrador de la aplicación con el menú “Carreras” desplegado.</i>	54
Figura 12. <i>Página principal del perfil administrador de la aplicación con el menú “Usuarios” desplegado.</i>	55
Figura 13. <i>Menú “Gestionar Usuarios”.</i>	56
Figura 14. <i>Menú “Anadir Usuario’.</i>	57
Figura 15. <i>Formulario Editar o Actualizar Usuario.</i>	57
Figura 16. <i>Usuario editado o actualizado correctamente.</i>	58
Figura 17. <i>Eliminar usuario.</i>	59

Figura 18. <i>Mensaje confirmación eliminar usuario.</i>	59
Figura 19. <i>Seleccionar carrera a consultar.</i>	60
Figura 20. <i>Seleccionar categoría.</i>	61
Figura 21. <i>Seleccionar página.</i>	61
Figura 22. <i>Vista Información consultada.</i>	62
Figura 23. <i>Ejemplos de filtros presentes 1.</i>	63
Figura 24. <i>Ejemplos de filtros presentes 2.</i>	63
Figura 25. <i>Ejemplos de aplicación de filtros a una página.</i>	64
Figura 26. <i>Seleccionar el apartado de “Gráficos”.</i>	65
Figura 27. <i>Graficar información 1.</i>	65
Figura 28. <i>Graficar información 2.</i>	66
Figura 29. <i>Vista del “Asistente inteligente IA”.</i>	67
Figura 30. <i>Ejemplo de preguntas para el “Asistente inteligente”.</i>	67
Figura 31. <i>Ejemplo de una consulta al “Asistente inteligente”.</i>	68

Resumen

Título: APLICATIVO WEB PARA LA CONSOLIDACION DE DATOS Y LA GESTION DE LA ACREDITACION ACADEMICA DE LOS PROGRAMAS UIS.*

Autor: Hordan Andrés Navarro Herrera, Luis Duvan Fernando Pinto Diaz.**

Palabras Clave: Software, webscraping, web.

Descripción: El proceso de acreditación institucional se debe realizar cada cierto tiempo por los claustros educativos que quieren conservar el galardón de alta calidad otorgado por el Ministerio de Educación Nacional (MEN) a la universidades del país, este proceso es algo complejo y largo de realizar pues se divide en 8 ítems a cumplir uno de ellos denominado Justificación que se enfoca en la pertinencia del programa, lo que quiere decir que se encarga de estudiar la oferta y demanda del programa académico en el ámbito económico, social, cultural, entre otros tanto a nivel local, nacional como internacional. Para esto se hace necesario la búsqueda de mucha información proveniente de diferentes fuentes lo que dificulta el desarrollo del proceso de forma rápida y eficiente. Actualmente existen muchas tecnologías que realizan la extracción de información de la web de forma automática lo que permitirá facilitar el proceso al menos en la obtención de los datos necesitados por el encargado del proceso. Para esto se desarrolló una aplicación web capaz de extraer y centralizar esta información en una base de datos que se estará actualizando automáticamente a cada cierto tiempo preprogramado con el fin de brindar la última información de interés y así reducir la complejidad del ítem de la pertinencia para el proceso de acreditación institucional.

* Trabajo de Grado

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería de Sistemas. Ingeniería de sistemas. Director: Carlos Alfonso Mantilla Duarte. M.Sc. en Estadística Aplicada. Codirector: Lola Xiomara Bautista Roza. PhD en Procesamiento de Imágenes y Señales

Abstract

Title: WEB APPLICATION FOR DATA CONSOLIDATION AND MANAGEMENT OF ACADEMIC ACCREDITATION OF UIS PROGRAMS.*

Author(s): Hordan Andrés Navarro Herrera, Luis Duvan Fernando Pinto Diaz.**

Key Words: Software, webscraping, web.

Description: The institutional accreditation process must be carried out from time to time by the educational cloisters that want to preserve the high quality award granted by the Ministry of National Education (MEN) to the country's universities, this process is somewhat complex and long to carry out since it is divided into eight items to fulfill one of them called Justification that focuses on the relevance of the program, which means that it is in charge of studying the offer and demand of the academic program in the economic, social, cultural field, among others both at the local, national and international. For this, it is necessary to search for a lot of information from different sources, which makes it difficult to develop the process quickly and efficiently. Currently there are many technologies that perform the extraction of information from the web automatically, which will facilitate the process, at least in obtaining the data needed by the person in charge of the process. For this, a web application was developed capable of extracting and centralizing this information in a database that will be automatically updated at pre-programmed times in order to provide the latest information of interest and thus reduce the complexity of the relevance item for the institutional accreditation process.

* Degree Work

**Faculty of Engineering Physique-Mechanics. School of Systems Engineering and Informatics. Director: Carlos Alfonso Mantilla Duarte. M.Sc. in Applied Statistics. Codirector: Lola Xiomara Bautista Rozo. PhD in Image and Signal Processing.

Introducción

La acreditación institucional es un proceso esencialmente académico el cual se configura como un mecanismo para la búsqueda de mayores niveles de calidad de los programas en estudio. es un proceso necesario llevado a cabo cada cierto tiempo por todos los claustros educativos que quieren conservar la certificación de alta calidad por parte del Ministerio de Educación Nacional bajo el decreto 1330 del 25 de julio de 2019 y la resolución 021795 de 2020 con el fin de mantenerse a la vanguardia con respecto a programas afines a nivel mundial y apuntar por una educación calidad en el país. La Universidad Industrial de Santander al igual que los demás claustros educativos del país se deben regir por los estándares de calidad que dicta el Consejo Nacional de Acreditación (CNA).

Para este proceso es necesario tener en cuenta ciertos requerimientos que se deben analizar y cumplir. La pertinencia que según la CNA consiste en “la actuación congruente, conveniente, coherente y adecuada a las condiciones y demandas sociales, culturales y ambientales del contexto en el que las instituciones y los programas académicos desarrollan sus labores formativas, académicas, docentes, científicas, culturales y de extensión”. se fundamenta principalmente en buscar las mejores condiciones de desarrollo de un programa teniendo en cuenta su entorno. Este análisis se debe justificar en el ámbito regional, nacional e internacional en correlación directa con la demanda laboral, la oferta educativa, empleabilidad, el entorno social, económico, entre otras cosas a tener en cuenta según los requerimientos, la demanda y especialidad de cada programa. Este proyecto basa sus esfuerzos en facilitar la obtención de esta enorme cantidad de información con el fin de proveer una herramienta para dicho proceso por medio del desarrollo de un aplicativo web y de tecnologías como el Web Scraping.

Mediante el uso de Web Scraping automático, el cual por medio de la creación de algoritmos o “pequeños softwares” y que tiene como objetivo extraer y almacenar datos y/o información de páginas web, enfocando sus alcances más en la conversión de datos sin estructurar como los encontrados generalmente en formato HTML u otro tipo en datos estructurados y que puedan ser guardados para posteriormente ser tratados en una base de datos central, esto facilita la extracción, obtención, procesamiento y el análisis de la información necesaria para llevar a cabo el proceso de forma más rápida. Teniendo en cuenta el alcance del problema se desarrolló un software simple, poderoso y capaz con una interfaz de visualización sencilla que no cansa la vista al usuario y que tiene por objeto obtener, clasificar y mostrar la información solicitada de forma ordenada para el profesional encargado de realizar dicho proceso. A continuación, se plasmó el trabajo realizado durante todo este tiempo y se explicó a detalle la metodología de desarrollo usada se describieron los casos de uso presentes que explican el funcionamiento del software, y se creó un manual de usuario que explica detalle cómo realizar las diferentes tareas o acciones en el software.

1. Planteamiento y Justificación del Problema

El Decreto 1330 del 25 de julio de 2019 y la resolución 021795 de 2020 definen los criterios necesarios para los procesos de acreditación de los diferentes programas académicos del país, este proceso requiere que las diferentes unidades académico administrativas que conforman la Universidad Industrial de Santander UIS lleven a cabo un proceso complejo de búsqueda y consolidación de la información que permita analizar la oferta local, nacional e incluso internacional para cada programa con denominación similar al que es objeto de estudio, esto con el fin de medir diferentes aspectos relacionados con los campos afines a los programas que ofrece la institución. El desarrollo de un aplicativo o herramienta de consulta que facilite la consolidación y la búsqueda de la información permitirá al usuario realizar tareas más ágiles y confiables para así optimizar de este modo las tareas asociadas al proceso de acreditación. Este proceso actualmente es realizado manualmente por los profesionales encargados de realizar este estudio sobre el programa y básicamente consiste en recopilar las fuentes de datos necesarias para la extracción de la información, seguido de revisar cada sitio para encontrar la información de interés y ver la forma de disponer de esta que muchas veces vienen en densas cantidades difícil de encontrar.

2. Objetivos

2.1 Objetivo General

Desarrollar una aplicación web, para la Universidad Industrial de Santander que facilite la obtención y consolidación de datos para la gestión del ítem “justificación del programa” enfocado a la pertinencia del programa especificado en el decreto 1330 del 25 de julio de 2019, con el fin de facilitar la preparación para el proceso de acreditación institucional para los diferentes programas ofrecidos por la universidad.

2.2 Objetivos Específicos

- Definir los lineamientos específicos de búsqueda de evidencia e indicadores de justificación de los programas.
- Desarrollar un aplicativo web que nos facilite la gestión de datos en los procesos de acreditación, tales como oferta y demanda de los diferentes programas académicos.
- Realizar análisis y clasificación de la información obtenida mediante un algoritmo de inteligencia artificial.
- Diseñar los reportes detallados sobre las consultas realizadas.

3. Propósito del Sistema

El propósito de este aplicativo fue el de diseñar e implementar una aplicación web que permita a los profesionales encargados de realizar el proceso de acreditación, obtener de forma automática por medio de la tecnología de webscraping y centralizado en el aplicativo la información que requieran para llevar a cabo dicho proceso para posteriormente ser tratada y/o manipulada para dichos fines.

Para esto se tendrá un administrador o superusuario el cual tendrá la capacidad de crear, editar ver y borrar las cuentas disponibles para el funcionamiento de la aplicación, este a su vez tendrá acceso y control total de la aplicación.

Y para los usuarios de la aplicación se les asignara una cuenta de acuerdo con su rol, estos tendrán que hacer el debido Login lo que les permitirá realizar las consultas en las páginas de preferencia, eliminar la información, descargarla y en general disponer de ella para su uso.

4. Marco de Referencia

4.1 SNIES

El Sistema Nacional de Información de la Educación Superior (SNIES) sitio web diseñado por el Ministerio de Educación Nacional y reglamentado mediante Decreto 1767 de 2006, es un sistema de información que permite llevar control, monitoreo, inspección y vigilancia del sector universitario del país. (SNIES Sistema Nacional de Información de la Educación Superior, 2017, 7 septiembre) Se encarga de recoger y organizar información sobresaliente sobre la educación superior para determinar su comportamiento y poder orientar a las Instituciones de Educación Superior en el proceso de mejoramiento a partir de la identificación de mejores prácticas.(M.E.N., 2017) Todas las universidades oficiales deben realizar sus reportes de información requerida por el Ministerio de Educación Nacional en esta página, siendo esta una de las principales fuentes de información que se tiene en este entorno, es posible conocer indicadores y en general el comportamiento a nivel nacional en el ámbito universitario de los programas afines a los ofrecidos por UIS. Una de las limitaciones que nos ofrece este sistema de información es que solo nos brinda información interna de las Página 3 de 13 universidades, haciendo que sea un punto de referencia en la obtención de información, pero insuficiente para la requerida para el proceso general de acreditación institucional.

4.2 Web Scraping

Web scraping también conocido como web harvesting es una técnica de extracción de datos que se encarga de extraer y almacenar datos de sitios o páginas web para analizarlos o utilizarlos fuera de este. Este se define como la creación de un agente o algoritmo para descargar, revisar y organizar datos de la web de una manera automatizada. (Broucke, 2018) El scraping automático,

en el cual se recurre a un software o un algoritmo que analiza de manera autónoma diferentes páginas web para extraer información y utiliza software especializado según el tipo y estructura de la página web y el contenido a extraer. Por medio de este raspado web se obtienen y se almacenan diversos tipos de información. El web scraping es muy usado en el ámbito del mundo de los datos financieros: es posible leer datos desde un sitio web externo, organizarlos en la forma que se desee y después analizarlos y procesarlos.

4.3 ¿Quiénes y para que están usando web scraping con inteligencia artificial?

El web Scraping ha sido una técnica que se ha venido desarrollando casi desde la invención del World Wide Web y a lo largo de los años se ha usado en diferentes tipos de investigaciones alrededor del mundo, pues gracias a su capacidad de obtener cantidades de datos con un solo algoritmo facilita el trabajo que a un humano le costaría demasiado tiempo realizar manualmente, son ejemplo de esto:

La naciente empresa hiQ con sede en San Francisco es un startup que se especializa en la venta de análisis de empleados por medio de la recopilación y el procesamiento de información de carácter y dominio público tal y como la que los mismos usuarios proveen a las redes sociales y demás. El web Scraping está siendo usado cada vez con mayor frecuencia en el análisis de recursos humanos, lo que no le gustó mucho a LinkedIn, quien por medio de los estrados judiciales ha intentado hasta ahora evitar que hiQ realice esta práctica en su página, aunque por ahora ha resultado de manera infructuosa, pues la información con la que hiQ trabaja es de carácter público. (Bennett, 2017)

Otro estudio en el cual también usan la información de carácter público obtenida mediante el Web Scraping que ofrecen las redes sociales en este caso los mensajes de Twitter, blog y otras redes sociales se usaron para crear un vasto conjunto de datos y este a su vez se utilizó para

construir un modelo predictivo de machine learning que intenta predecir e identificar patrones de depresión y pensamientos suicidas lo que significaría una herramienta invaluable para salvar vidas. (Sprajcar, 2016)

“El proyecto de mil millones de precios” es un artículo que usó precios en línea para el cálculo, la medición e investigación, se utilizó el Web Scraping para recopilar un conjunto de datos de información de precios en línea que fue utilizado para construir un índice de precios diario robusto para muchos países. En particular, lo que se muestra es el cómo se pueden utilizar los precios en línea para construir índices de precios en lapsos de tiempo ya sea horas, días etc. En varios países y con esto evitar sesgos de medición que distorsionan la evidencia de rigidez de precios y precios relativos internacionales. (Cavallo, 2016).

5. Tecnologías a Usar

5.1 Python

Es un lenguaje de programación interpretado de alto nivel, dinámico, versátil, multiplataforma, y multiparadigma de código abierto que se caracteriza por su código legible y limpio lo que permite su uso en amplios contextos y propósitos clasificándose constantemente como uno de los lenguajes de programación más populares del mundo. (About Python, s.f.) Es usado para una gran variedad de tareas, como el desarrollo de aplicaciones web, el procesamiento de datos y la automatización y además Python también cuenta con una gran comunidad de desarrolladores y una amplia variedad de bibliotecas y herramientas para ayudar en el desarrollo de proyectos entre sus principales características encontramos:

- Lenguaje de alto nivel: esto significa que se parece más al lenguaje humano y es fácil de leer y escribir.
- Interpretado: lo que significa que el código se ejecuta línea por línea en lugar de ser compilado antes de ejecutarse.
- Sintaxis legible: La sintaxis de Python es clara y legible, lo que facilita la escritura y la lectura del código.
- Tipado dinámico: lo que significa que no se requiere especificar el tipo de una variable al declararla.
- Gran comunidad: Python tiene una gran comunidad de desarrolladores, lo que se traduce a una gran cantidad de recursos y bibliotecas disponibles.
- Multiparadigma: esto significa que es compatible con varios estilos de programación, como programación orientada a objetos, programación funcional y programación estructurada.
- Portabilidad: Python se ejecuta en una variedad de plataformas, como Windows, MacOS, Linux, y se puede utilizar en diferentes dispositivos, como servidores, computadoras personales, dispositivos móviles.
- Riqueza de librerías: Python cuenta con un gran número de librerías y módulos que amplían su funcionalidad y facilitan el desarrollo de aplicaciones en distintas áreas, como ciencia de datos, inteligencia artificial, automatización, entre otras.

5.2 JavaScript

JavaScript conocido comúnmente como JS es un lenguaje de programación interpretado, de alto nivel, que utilizado principalmente para desarrollar contenido interactivo y dinámico en sitios web. (¿Qué es JavaScript?, s.f.). Con JavaScript, los desarrolladores pueden agregar una

variedad de características interactivas a las páginas web, como formularios de entrada, menús desplegables, animaciones y entre otras. Es un lenguaje muy versátil y se puede utilizar tanto en el lado del cliente (ósea en el navegador) como en el lado del servidor (a través del entorno de Node.js). Es uno de los lenguajes de programación más populares en el mundo y es esencial para cualquier desarrollador web que desee crear sitios web interactivos y modernos.

5.3 PHP

PHP es un lenguaje de programación de código abierto, diseñado especialmente para el desarrollo web. (¿Qué es PHP?, s.f.) Se utiliza para generar contenido dinámico en páginas web, permitiendo a los desarrolladores crear sitios web interactivos y aplicaciones en línea.

PHP es un lenguaje de script del lado del servidor, lo que significa que el código PHP se ejecuta en el servidor antes de que la página web sea enviada como respuesta al cliente. Esto permite a los desarrolladores crear aplicaciones web con una gran cantidad de funcionalidad, como el manejo de sesiones de usuario, el almacenamiento de información en una base de datos y la generación de contenido dinámico. Este lenguaje se utiliza para una variedad de usos y aplicaciones, desde sitios web simples hasta sistemas de gestión de contenido (CMS) y plataformas de comercio electrónico muy complejas. Es compatible con una amplia variedad de bases de datos y sistemas operativos, y se integra fácilmente con HTML, CSS y JavaScript.

5.4 HTML5

HTML (Hypertext Markup Language) es un lenguaje de marcas o marcado utilizado para crear páginas web. Es el estándar principal para la creación de páginas web y se utiliza junto con otros lenguajes como JavaScript y CSS para crear sitios web interactivos y dinámicos. HTML está compuesto por etiquetas y atributos que se utilizan para definir la estructura y el contenido de una página web. (¿Qué es HTML?, s.f.). HTML5 es la quinta versión del lenguaje e incluye nuevas

etiquetas y características que mejoran la funcionalidad de las páginas web, como el soporte para videos y audio integrados, el uso de elementos de formulario más avanzados y la capacidad de almacenar datos en el lado del cliente. Además, HTML5 es compatible con dispositivos móviles y tiene una mayor accesibilidad.

5.5 CSS

CSS (Cascading Style Sheets) en español hojas de Estilo en Cascada, es un lenguaje de diseño utilizado para describir el aspecto visual de un documento escrito en un lenguaje de marcas, como HTML. (¿Qué es CSS?, s.f.). Con CSS, los desarrolladores pueden aplicar estilos (colores, tipografías, espaciado, etc.) a los elementos de una página web de manera consistente y separada del contenido de esa página. Esto permite un mayor control sobre el diseño de la página y facilita la modificación y actualización de los estilos en el futuro. CSS también permite a los desarrolladores crear diseños adaptativos que se ajustan automáticamente a diferentes pantallas y dispositivos conocido como responsive.

5.6 Django

Django es un framework de desarrollo web de código abierto creado para el lenguaje de programación Python. (Why Django. Django overview, s.f.) Fue diseñado para ayudar a los desarrolladores a crear aplicaciones web de manera rápida y sencilla, proporcionando una estructura sólida para el manejo de la base de datos, la creación de formularios, la validación de datos y otras tareas comunes en el desarrollo web. Django utiliza el patrón de arquitectura Modelo-Vista-Controlador (MVC) y se enfoca en la reutilización y la extensibilidad, permitiendo a los desarrolladores crear aplicaciones web complejas utilizando componentes previamente construidos, entre las características más destacadas de Django se tenemos:

- Posee un administrador de sitios web incorporado que permite a los administradores gestionar el contenido de un sitio web de manera eficiente.
- Tiene un sistema de plantillas para separar la lógica de presentación de los datos.
- Dispone de un sistema de autenticación y autorización robusto.
- Es muy versátil ya que cuenta con integración con bases de datos populares, como PostgreSQL, MySQL y SQLite.

5.7 Selenium Webdriver

Selenium es una suite de herramientas de automatización de pruebas para aplicaciones web conocido como webscraping. (Webdriver. Webdriver Selenium, s.f.). Fue diseñado para automatizar pruebas funcionales y de aceptación en navegadores web, lo que permite a los desarrolladores y a los equipos de pruebas automatizar pruebas de navegación, interacción con formularios, verificación de contenido y otras tareas comunes en el proceso de pruebas de una aplicación web. Selenium WebDriver es una API que permite a los desarrolladores escribir scripts de prueba en varios lenguajes de programación, incluyendo Python, Java, C# y Ruby, para automatizar la interacción con un navegador web. En resumen, Selenium es una suite de herramientas de automatización de pruebas para aplicaciones web que permite a los desarrolladores y equipos de pruebas automatizar pruebas funcionales y de aceptación en navegadores web, permitiendo una mayor eficiencia en el proceso de pruebas y detectando errores de manera temprana.

5.8 Scrapy

Scrapy es un marco de rastreo web de código abierto y colaborativo para Python. (Scrapy 2.7 documentation, s.f.). Se utiliza para extraer datos de sitios web y se puede utilizar para una amplia variedad de propósitos, como minería de datos, procesamiento de datos o archivo histórico

webscraping entre otros. Scrapy proporciona una forma integrada de seguir enlaces y extraer datos de sitios web, y se puede ampliar fácilmente para manejar comportamientos personalizados o extracción de datos personalizados.

5.9 API

API (Application Programming Interface) que en español significa interfaz de programación de aplicaciones, es un conjunto de reglas y protocolos para construir y desarrollar programas que interactúen entre dos o más sistemas. Una API específica define cómo componentes de software deben interactuar y permite que diferentes aplicaciones se comuniquen entre sí. Una API puede ser de diferentes tipos, pero comúnmente se refiere a una API de software que se utiliza para acceder a un sistema de bases de datos o un servicio web.

5.10 BeautifulSoup

Beautiful Soup es una biblioteca diseñada especialmente para Python utilizada para el web scraping que consiste en extraer datos de sitios web. (Beautiful Soup: We called him Tortoise because he taught u, s.f.). Es una herramienta de análisis de datos utilizada para navegar, buscar y modificar la estructura de un documento HTML o XML. BeautifulSoup proporciona una interfaz amigable para buscar y navegar en el árbol de nodos generado a partir de un documento HTML o XML, lo que permite a los desarrolladores acceder fácilmente a los datos específicos que necesitan. Es comúnmente utilizado en conjunto con otras herramientas de web scraping como Scrapy o Selenium.

5.11 Modelo Vista-Controlador

El patrón de diseño Modelo-Vista-Controlador (MVC) es un patrón de arquitectura de software comúnmente utilizado en desarrollo de aplicaciones. Se utiliza para separar la lógica de negocio de una aplicación de su interfaz de usuario y su control de flujo.

- El Modelo representa la lógica de negocio y los datos de la aplicación. Es responsable de acceder a los datos, realizar cálculos y lógica, y notificar a la vista y al controlador de cualquier cambio en los datos.
- La Vista representa la interfaz de usuario de la aplicación. Es responsable de mostrar los datos al usuario y recibir las acciones del usuario.
- El Controlador es el intermediario entre la vista y el modelo. Es responsable de recibir acciones del usuario de la vista, actualizar el modelo y notificar a la vista de cualquier cambio en los datos.

La separación de responsabilidades entre estos tres componentes permite una mayor flexibilidad y escalabilidad en el desarrollo de aplicaciones, ya que cada componente puede ser desarrollado y probado independientemente.

5.12 Django Token Authentication

Django Token Authentication es un método para autenticar a los usuarios en una aplicación Django mediante el uso de tokens. En lugar de utilizar credenciales tradicionales como nombres de usuario y contraseñas, los usuarios obtienen tokens después de iniciar sesión. Luego pueden enviar ese token en cada solicitud posterior para autenticarse. (Authentication - Django REST framework, s.f.). Django proporciona una serie de herramientas para trabajar con tokens de autenticación, como la creación, almacenamiento y verificación de tokens. Los desarrolladores también pueden crear sus propias soluciones personalizadas utilizando los mecanismos de autenticación de Django. Los tokens son especialmente útiles en aplicaciones móviles y de cliente-servidor, ya que permiten a los usuarios iniciar sesión una sola vez y luego mantener su sesión iniciada en varias solicitudes, sin tener que volver a ingresar sus credenciales.

Django Token Authentication se puede implementar utilizando el paquete de Django Rest Framework (DRF), que proporciona varias clases de autenticación que se pueden agregar fácilmente a vistas o conjuntos de vistas para manejar la autenticación basada en token. Algunos de los más comunes incluyen TokenAuthentication y JSONWebTokenAuthentication.

5.13 Django Rest Framework

Django Rest Framework (DRF) es un paquete de software de código abierto para Django que proporciona una fácil implementación de la arquitectura REST para aplicaciones web. (Home - Django REST framework, s.f.). Es una herramienta para crear servicios web basados en REST utilizando el marco de trabajo de Django. Este proporciona una gran cantidad de funcionalidades de forma nativa, como la serialización, autenticación, paginación, entre otras. Además, soporta varios formatos como JSON, XML, YAML, y permite una fácil personalización y extensión del framework.

DRF también proporciona una interfaz de navegación de la API, lo que permite una fácil prueba y depuración de la API y también soporta versionamiento de la API. Es ampliamente utilizado por los desarrolladores, ya que les permite crear APIs robustas y escalables rápida y fácilmente, al mismo tiempo que se adhiere a los principios de la arquitectura REST. Es una elección popular entre los desarrolladores que utilizan Django para construir aplicaciones web y desean exponer sus datos y funcionalidad a través de una API.

5.14 Arquitectura De Microservicios

Los Microservicios son un enfoque de arquitectura de software que consiste en dividir una aplicación en pequeños servicios independientes que se ejecutan de forma autónoma. Cada servicio se enfoca en una tarea específica y se comunica con otros servicios a través de una interfaz

establecida. Los Microservicios tienen varias ventajas sobre las aplicaciones monolíticas tradicionales, entre ellas tenemos:

- Escalabilidad: los servicios se pueden escalar de forma independiente según sea necesario.
- Flexibilidad: los servicios pueden desarrollarse y probarse independientemente, lo que facilita la creación de aplicaciones con un equipo de desarrollo más grande.
- Facilidad de mantenimiento: los errores en un servicio no afectan a otros servicios, lo que facilita la depuración y el mantenimiento de la aplicación.
- Facilidad de reutilización: los servicios pueden ser reutilizados en otras aplicaciones.

Sin embargo, también tienen desventajas, como la complejidad adicional en la gestión y monitoreo de servicios, y la necesidad de una infraestructura adecuada para el despliegue y comunicación de servicios.

5.15 JSON

JSON (JavaScript Object Notation) en español notación de objeto de JavaScript, es un formato de texto ligero para el intercambio de datos. Es un estándar abierto que se utiliza para representar datos estructurados en la sintaxis de objetos de JavaScript, es fácil de leer y escribir para humanos y fácil de analizar y generar para las computadoras. (Introducción a JSON, s.f.).

JSON utiliza una sintaxis similar a la de JavaScript para representar los datos. Los datos se organizan en pares clave-valor y se separan mediante comas. Los valores pueden ser números, cadenas, booleanos, objetos, matrices o valores especiales NULL. Los objetos y matrices se delimitan mediante llaves {} y [] respectivamente. JSON se ha convertido en uno de los formatos de intercambio de datos más populares en el desarrollo web, especialmente en el contexto de la

comunicación entre aplicaciones cliente-servidor. Es ampliamente utilizado en RESTful APIs, y es compatible con una amplia variedad de lenguajes de programación, como JavaScript, Python, Java, Ruby y muchos otros

5.16 Minería De Datos

Minería de Datos La minería de datos es el proceso de incursionar y explorar las grandes cantidades de datos para descubrir relaciones desconocidas y predecir tendencias futuras. Conocido algunas veces como "descubrimiento de conocimientos en bases de datos", el término "minería de datos" no se usó sino hasta la década de los 90 para referirse a esta actividad. Sus bases están comprendidas por tres disciplinas científicas entrelazadas: estadística, inteligencia artificial y machine learning. Lo que era antiguo es nuevo otra vez, ya que la minería de datos continúa evolucionando para igualar el ritmo del potencial sin límites del big data y poder de cómputo asequible a nuestros días. (Orallo, 2004).

En los últimos años, los avances tecnológicos se han reflejado en la capacidad y la velocidad de procesamiento que nos han permitido ir más allá de las prácticas manuales, tediosas y que toman mucho tiempo al análisis de datos rápido, fácil y automatizado. Cuanto más complejos son los conjuntos de datos recopilados, mayor es el potencial que hay para descubrir patrones, insights, e información relevante. Los comerciantes detallistas, bancos, fabricantes, proveedores de telecomunicaciones y aseguradoras, entre muchos otros, utilizan la minería de datos para descubrir relaciones entre todas las cosas, desde precios, promociones y demografía hasta la forma en que la economía, el riesgo, la competencia y los medios sociales afectan sus modelos de negocios, ingresos, operaciones y relaciones con clientes.

5.17 Inteligencia Artificial

Inteligencia Artificial La Inteligencia Artificial se podría definir como la combinación de algoritmos creados con el fin de crear acciones o máquinas que presenten las mismas capacidades que un ser humano. (¿Qué es la inteligencia artificial (IA)?, s.f.). Dichos algoritmos dotan a los computadores o máquinas de la capacidad de identificar patrones en datos masivos y elaborar diferentes tipos de acciones y/o predicciones; en un sentido más amplio la IA es una ciencia dirigida al diseño y creación de dispositivos, máquinas, carros etc. que implementen tareas propias como lo haría un ser humano dotado de inteligencia. (Fleitas, 2017)

5.18 Docker

Docker es una plataforma de contenedores que permite empaquetar y distribuir aplicaciones de manera rápida y eficiente. Los contenedores son una forma de virtualización ligera que permite ejecutar una aplicación en un entorno aislado, con todas las dependencias y configuraciones necesarias. Esto significa que una aplicación empaquetada en un contenedor puede ser ejecutada en cualquier sistema que tenga Docker instalado, sin importar las diferencias en configuraciones de sistema o versiones de dependencias. (About Docker, s.f.).

Docker tiene varias características útiles, como:

- **Imágenes:** Una imagen de Docker es un paquete de software que contiene todo lo necesario para ejecutar una aplicación, incluyendo el código fuente, las dependencias, las configuraciones y los recursos.
- **Contenedores:** Un contenedores una instancia de una imagen de Docker que se está ejecutando. Puedes tener varios contenedores corriendo a partir de la misma imagen.
- **Registros:** Docker Hub es el registro de imágenes de Docker oficial, donde se pueden encontrar y descargar millones de imágenes de aplicaciones populares.

- Compose: Es una herramienta para definir y ejecutar múltiples contenedores de manera fácil y sencilla.
- Swarm: Es una herramienta de Docker para crear y administrar clústeres de contenedores.

Con Docker, se puede desarrollar, probar y desplegar aplicaciones de manera rápida y fácil, ya que el entorno de ejecución es consistente en todos los sistemas. Esto ayuda a reducir problemas de compatibilidad y aumenta la eficiencia del desarrollo y el despliegue.

5.19 Crontab

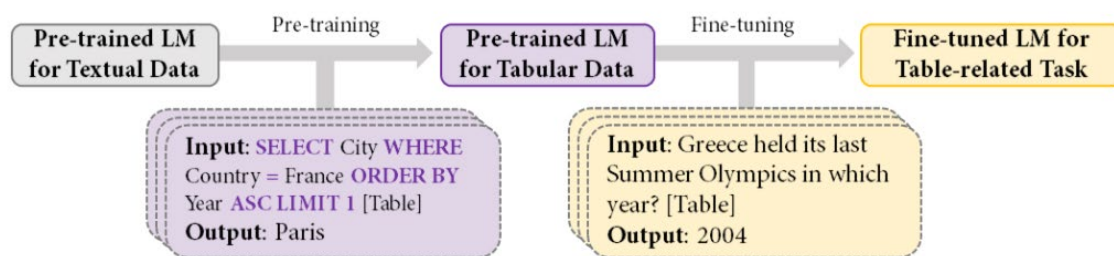
Python-crontab es un módulo de Python que permite programar tareas (como scripts de Python) para ejecutarse en un horario específico utilizando el sistema cron de Linux. Con este módulo, puedes agregar, eliminar o listar tareas programadas en el sistema y personalizar la ejecución de tus scripts automatizando tareas diarias. Este módulo se basa en la librería Python-crontab de Python, para interactuar con el sistema cron del sistema operativo. (python-crontab, s.f.)

5.20 TAPEX

Recientemente se han logrado grandes avances en el preentrenamiento de modelos de lenguaje usando datos estructurados no tabulares, sin embargo, es aún un gran reto el preentrenamiento de estos modelos usando datos tabulares. TAPEX (Table Pre-training via Execution) o Table Pre-training via Learning a Neural SQL Executor en español Reentrenamiento previo de tablas, es un enfoque de preentrenamiento de modelos con datos tabulares, el cual entrena previamente un modelo BART para resolver consultas SQL sintéticas, para posteriormente responder preguntas en lenguaje natural relacionadas con datos tabulares, así como para realizar la verificación de datos de tablas. (Qian, 2022)

El objetivo principal de TAPEX es desarrollar un modelo para reproducir el proceso de ejecución de consultas SQL en una tabla. La premisa es que, si un modelo es capaz de ejecutar las consultas SQL de forma precisa, entonces es probable que tenga una comprensión completa de las estructuras de tablas y tenga una tendencia a aprender sobre esas estructuras.

Figura 1. Diagrama de funcionamiento del modelo de inteligencia artificial Table Pre-training via Learning a Neural {SQL} Executor.



Nota. El grafico muestra a grandes rasgos el paso a paso del funcionamiento del modelo de inteligencia artificial usado. Tomado de *International Conference on Learning Representations*. por Qian Liu, Bei Chen, Jiaqi Guo, Morteza Ziyadi, Zeqi Lin, Weizhu Chen, Jian-Guang Lou.

5.21 Docker Compose

Docker Compose es una herramienta muy útil que nos permite definir y correr aplicaciones compuestas por múltiples contenedores Docker, puedes especificar los servicios que conforman tu aplicación en un único archivo de configuración y luego ejecutar un solo comando para crear y ejecutar todos los servicios. Esto facilita la gestión y configuración de aplicaciones complejas compuestas por varios contenedores divididos en microservicios.

5.22 GitHub

GitHub es una plataforma web para alojar y colaborar en proyectos de software utilizando el sistema de control de versiones Git. Con GitHub, los desarrolladores pueden guardar y compartir su código, trabajar en proyectos con otros desarrolladores, rastrear y solucionar problemas, y mucho más. GitHub se ha convertido en una de las plataformas más populares para el desarrollo de software y la colaboración en proyectos open source. (About GitHub, s.f.)

6. Metodología

6.1 Metodología De Desarrollo SCRUM

Para el actual trabajo se optó por el tipo de metodología de desarrollo ágil conocida como SCRUM, esta es mayormente utilizada para el desarrollo de proyectos, especialmente de software como en este caso. Se basa en un enfoque iterativo e incremental, donde un equipo multidisciplinario de desarrollo trabaja en sprints cortos para completar determinadas tareas y alcanzar metas específicas. El proceso está dirigido por un Scrum Máster, quien facilita el trabajo del equipo y asegura que se sigan las prácticas de Scrum al pie de la letra. (Monte Galiano, 2016).

El proceso de Scrum se divide en tres roles principales:

- **Product Owner:** Es responsable de establecer la prioridad de las tareas y de asegurar que el trabajo realizado esté en línea con las metas del proyecto, para este caso ambos actuaremos de forma crítica como Product Owners y para esto hemos dispuesto de un planificador en Jira que nos permitió crear los Sprints y repartirlos de forma coherente en el tiempo y con lo dispuesto en nuestro cronograma de actividades.

- **Scrum Master:** Es el responsable de facilitar el trabajo del equipo, asegurando que se sigan las prácticas de Scrum y resolviendo cualquier problema que pueda surgir, en este ámbito nos valimos de toda la ayuda que nos brindó el profesor en la guía para el desarrollo y encaminamiento del proyecto.

- **Equipo de desarrollo:** Son los encargados de realizar el trabajo y completar las tareas asignadas en cada sprint, para este caso ambos anteriormente habíamos definido nuestros Sprints con determinadas tareas las cuales fuimos ejecutando y cumpliendo de acuerdo con el cronograma.

El proceso de Scrum se divide en ciclos de trabajo llamados Sprints. Cada sprint tiene una duración de 2 a 4 semanas y tiene tres fases:

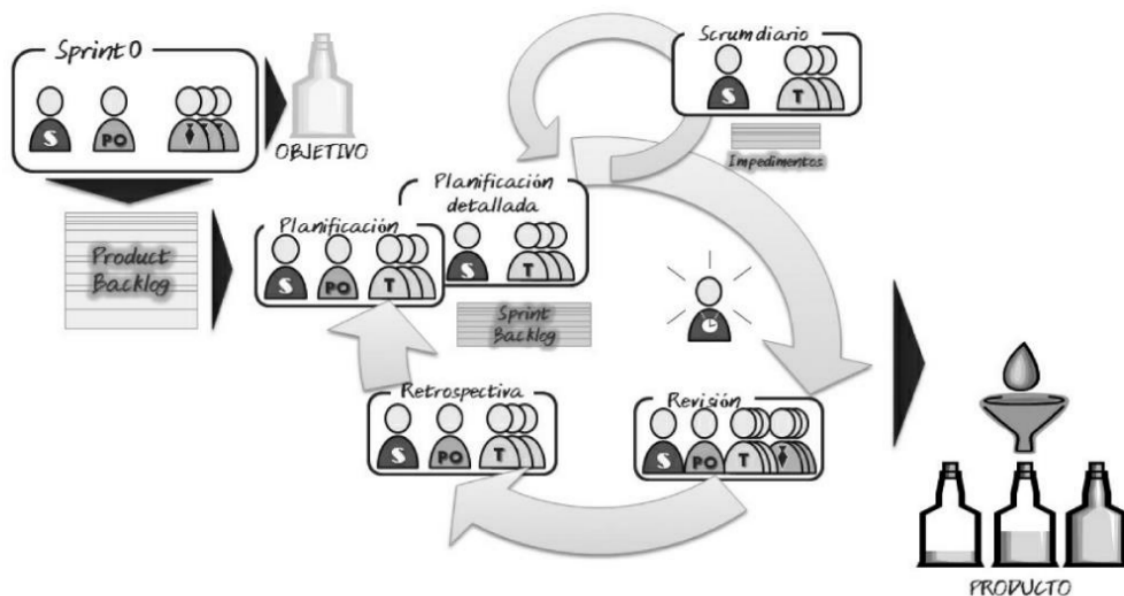
- **Planificación:** en esta fase se planeó el trabajo que se va a realizar en el sprint siguiente, se establecen metas y se asignan tareas a cada uno. Para esto se debe reunir al equipo de desarrolladores, y cualquier otra persona necesaria para completar la aplicación web. El equipo debe incluir un Scrum Máster y un Product Owner. Esta tarea se realizó cada 15 días en donde se seleccionaban las tareas del Product Backlog que se completarían durante el Sprint, y se asignaban responsabilidades y plazos a cada tarea, revisábamos que tan realistas estaban los objetivos a cumplir, si se había cumplido lo esperado para cada sprint y basándose en lo aprendido durante la revisión y retroalimentación del Sprint anterior si habían surgido nuevas características a añadir y así actualizar y generar nuevos Sprints.

- **Desarrollo:** en esta fase el equipo trabaja en las tareas asignadas y completa el trabajo planificado. Durante el Sprint, trabajamos en las tareas planificadas, utilizando la técnica de "daily Scrum" para mantenemos en sincronía y asegurar que se está avanzando según lo

planificado, el equipo puede usar herramientas de seguimiento de proyectos para registrar el progreso y la compleción de las tareas en nuestro caso jira.

- **Revisión y Retrospectiva:** en esta fase se revisa el trabajo realizado y se discuten los problemas y soluciones. Al final del Sprint, nos reunimos para revisar el trabajo completado y realizar retroalimentación del Product Owner en este caso nosotros mismos. En estas reuniones podíamos demostrar las funciones y características completadas durante el Sprint, y recibir comentarios y sugerencias de mejora entre nosotros.

Figura 2. El ciclo SCRUM.



Nota. El grafico muestra una representación de la distribución de tareas en el tiempo en la metodología de desarrollo SCRUM. Tomado de *Métodos Ágiles*. Anaya Multimedia, por Lasa Gómez, Carmen, Álvarez García, Alonso, & Heras del Dedo, Rafael de las. (2017).

Este proyecto se estructuró y dividió en 5 fases importantes las cuales fueron traducidas a sprint para su posterior desarrollo:

6.1.1 Fase I. Investigación.

Se definieron los lineamientos de búsqueda y los programas académicos para la identificación y recolección de datos necesarios teniendo en cuenta los sitios de búsqueda estipulados por los diferentes grupos focales encargados de cada una de las carreras. Para esto se debe tener conocimiento de los indicadores apropiados para el estudio de la acreditación en la justificación de cada programa académico.

6.1.2 Fase II. Análisis de Requerimientos.

Se establecieron los requerimientos funcionales y no funcionales para el desarrollo del aplicativo web, se identificaron los casos de uso para la justa realización de los requerimientos funcionales.

6.1.3 Fase III. Diseño.

En esta fase se diseñaron los casos de uso descritos en la fase anterior, se realizó el diagrama entidad relación de la base de datos para el aplicativo web, la arquitectura de la aplicación de acuerdo con los requerimientos establecidos en la fase anterior y se diseñó la parte gráfica de la aplicación web.

6.1.4 Fase IV. Desarrollo.

En esta etapa se llevó a cabo la implementación de todas las soluciones propuestas de las fases anteriores, se procedió a escribir el código fuente del aplicativo web indispensable para el proyecto y todas las configuraciones previas necesarias para su funcionamiento tales como la puesta en marcha del servidor, el correcto funcionamiento de la base de datos, la codificación de la interfaz gráfica (Front-end), la codificación lógica del aplicativo web (Back-end) y la

implementación de un algoritmo de IA que le permite al usuario escribir en lenguaje natural preguntas relacionadas al contenido de las tablas y obtener una respuesta.

6.1.5 Fase V. Prueba y Despliegue.

En esta última fase se verificó el funcionamiento del prototipo obtenido para eliminar los errores que fueron apareciendo en las pruebas realizadas y se terminó con el despliegue de la solución obtenida y el software terminado.

7. Modelos del Sistema

Los casos de uso son un tipo de modelo de sistema que se utiliza en la ingeniería de sistemas para identificar y describir los diferentes escenarios en los que un sistema será utilizado. Estos modelos son una forma de documentar las interacciones entre los usuarios y el sistema, y se utilizan para validar los requisitos, las funcionalidades del sistema entre otros. La elaboración de los casos de uso implica establecer los límites del sistema, es decir, distinguir las acciones que realiza el sistema de las que se realizan en su entorno. Es importante destacar que los actores se encuentran fuera de los límites del sistema, mientras que los casos de uso se sitúan dentro de estos límites.

7.1 Descripción de Casos de Uso

Los modelos de casos de uso presentes en el desarrollo de este aplicativo son los siguientes:

7.1.1 Caso de Uso 1: Hacer Login en la aplicación (Administrador).

Escenario caso de uso 1: En un principio la aplicación tendrá solo una cuenta administradora por defecto creada con antelación para el funcionamiento correcto del aplicativo pues será quien administre funcionamiento cuentas y en general la aplicación. Para esto se le pedirá al administrador que inicie sesión.

Actores participantes: Administrador.

Flujo de eventos:

1. El administrador abre el aplicativo web en un computador y le aparece opción de “Login”.
2. El administrador selecciona esta opción.
3. La aplicación le muestra los campos de correo y/o usuario y contraseña.
4. La aplicación valida que los campos de correo y contraseña sean correctos.
5. La aplicación lo redirige a la pantalla principal. Si los datos no concuerdan, la aplicación muestra el siguiente mensaje: “Datos de ingreso incorrectos”.

Condiciones de Entrada: El usuario debe haber sido registrado en la aplicación anteriormente por el administrador.

Condiciones de Salida: El usuario ha realizado su Login de manera exitosa.

7.1.2 Caso de Uso 2: El administrador registra o crea las cuentas de los usuarios (Administrador).

Escenario caso de uso 2: Cada usuario será registrado en la aplicación por el administrador; al momento de hacerlo se le asignará un rol dependiendo de su dependencia o escuela. El administrador debe disponer de los datos y el rol de la persona a la cual le va a crear la cuenta.

Actores participantes: Administrador.

Flujo de eventos:

1. El administrador abre la aplicación en su laptop o PC y hace su respectivo Login le aparece el formulario de registro.
2. La aplicación le muestra los campos de (Rol, Escuela, Documento, Nombre completo, correo, celular, contraseña)

3. La aplicación valida que no se haya registrado anteriormente.
4. La aplicación le mostrará un mensaje que dice: “Se ha registrado correctamente al usuario”.

7.1.3 *Caso de Uso 3:* Hacer Login en la Aplicación (Usuarios).

Escenario de casos de uso 3: El usuario debe disponer del nombre de usuario y contraseña para poder realizar el respectivo acceso.

Actores participantes: usuarios.

Flujo de eventos:

1. El usuario abre el aplicativo web en un computador y le aparece opción de “Login”.
2. El usuario escoge esta opción.
3. La aplicación le muestra los campos de correo y/o usuario y contraseña.
4. La aplicación valida que los campos de correo y contraseña sean correctos.
5. La aplicación lo redirige a la pantalla principal. Si los datos no concuerdan, la aplicación muestra el siguiente mensaje: “Datos de ingreso incorrectos”.

Condiciones de Entrada: El usuario debe haber sido registrado en la aplicación anteriormente por el administrador.

Condiciones de Salida: El usuario ha realizado su Login de manera exitosa.

7.1.4 *Caso de Uso 4:* Realizar una Consulta a una Página (Usuarios).

Escenario de casos de uso 4: El usuario hace uso del software para lo cual realizara una búsqueda en el aplicativo

Flujo de eventos

1. El usuario abre el aplicativo web en un computador y le aparece opción de “Login”.
2. El usuario introduce los datos para realizar el respectivo Login.

3. La aplicación se redirige al menú principal y muestra los datos de usuario.
4. El usuario se dirige directamente al módulo donde se encuentra el listado de páginas, selecciona una de ellas para que luego se le muestren los datos en pantalla consultados.

Condiciones de Entrada: El usuario debe haber sido registrado en la aplicación anteriormente por el administrador.

Condiciones de Salida: El usuario ha realizado su Login y respectiva consulta de manera exitosa.

7.1.5 Caso de Uso 5: El administrador edita un perfil de usuario (Administrador).

Escenario de casos de uso 5: El administrador debe poseer la información necesaria para poder realizar los respectivos cambios al perfil de usuario

Actores participantes: Administrador.

Flujo de eventos:

1. El administrador abre el aplicativo web en un PC le aparece opción de “Login” y debe acceder con su perfil de administrador.
2. El administrador se dirige a la sección que contiene las opciones para editar perfiles de usuario y da clic allí.
3. La aplicación le muestra los campos que habilitados para editar.
4. El administrador realiza las respectivas adecuación o cambios a la cuenta y le da en guardar.
5. La aplicación le mostrara un mensaje de “cambios guardados exitosamente” para confirmar que los cambios se realizaron exitosamente.

Condiciones de Entrada: El administrador debe iniciar sesión con su respectiva cuenta.

Condiciones de Salida: El administrador ha realizado los cambios de manera exitosa.

7.1.6 Caso de Uso 6: El administrador elimina un perfil de usuario (Administrador).

Escenario de casos de uso 6: El administrador debe poseer la información necesaria para poder eliminar al respectivo perfil de usuario.

Actores participantes: Administrador.

Flujo de eventos:

1. El administrador abre el aplicativo web en un PC le aparece opción de “Login” y debe acceder con su respectivo perfil de administrador.
2. El administrador se dirige a la sección eliminar perfil de usuario y da clic allí.
3. La aplicación le muestra la opción para eliminar cuenta que aparece al lado de cada cuenta de usuario.
4. El administrador da clic sobre la cuenta que quiere eliminar y le da en guardar.
5. La aplicación le mostrara un mensaje de “cambios guardados exitosamente” para confirmar que los cambios se realizaron exitosamente.

Condiciones de Entrada: El administrador debe iniciar sesión con su respectiva cuenta.

Condiciones de Salida: El administrador ha realizado la edición de manera exitosa.

7.1.7 Caso de Uso 7: Realizar una Grafica a una Página Consultada (Usuarios).

Escenario de casos de uso 7: El usuario hace uso del software para lo cual realizará una búsqueda en el aplicativo seleccionando la página y luego bajo unos determinados filtros se procederá a realizar la respectiva grafica.

Flujo de eventos

1. El usuario abre el aplicativo web en un computador y le aparece opción de “Login” y debe introducir los datos para realizar el respectivo loggin.

2. Una vez allí debe seleccionar la página de la cual desea obtener la información y darle en consultar.

3. Debajo de esto aparecerá la opción de graficar con determinados filtros se deben escoger los necesarios para poder mostrar la grafica

4. Después se debe dar clic en la opción graficar para posteriormente verse en pantalla la respectiva grafica

Condiciones de Entrada: El usuario debe haber sido registrado en la aplicación anteriormente por el administrador con su respectivo rol.

Condiciones de Salida: El usuario ha consultado la respectiva gráfica de su interés.

7.1.8 Caso de Uso 8: Realizar una Consulta con el asistente de IA (Usuarios, Administrador).

Escenario de casos de uso 8: El usuario hace uso del software para lo cual realizará una búsqueda en la sección del asistente de Inteligencia

Artificial.

Flujo de eventos

1. El usuario abre el aplicativo web en un computador y le aparece opción de “Login” y debe introducir los datos para realizar el respectivo login.

2. Una vez allí debe seleccionar la sección de búsqueda del asistente de IA.

3. Una vez allí y en lenguaje natural el superusuario realizara la búsqueda

4. Después se debe dar clic en la opción de consultar para luego recibir la respuesta por parte del asistente.

Condiciones de Entrada: El usuario debe haber sido registrado en la aplicación anteriormente por el administrador con su respectivo rol.

Condiciones de Salida: El usuario ha consultado la respectiva información de su interés.

Los anteriores son los principales casos de uso que se tuvieron en cuenta a la hora del desarrollo y que pudimos identificar de las funcionalidades del sistema, sin embargo, en caso de presentarse un caso de uso no mencionado para cualquiera de los usuarios del sistema este se podrá identificar al final en la sección “diseño” en la que se retratan a detalle las funcionalidades plausibles del sistema.

8. Análisis Diseño e Implementación

En este apartado encontraremos el análisis completo del software, como se desarrolló las limitaciones que presenta en sus funcionalidades y de que consta.

8.1 Análisis e Implementación

Esta versión del software fue desarrollada bajo la arquitectura de microservicios con el objetivo de hacer escalable el proyecto desarrollando interfaces simples para el usuario el cual cuenta con 3 microservicios por separado los cuales fueron desarrollados en Python, JavaScript, PHP, entre otros.

El backend es la parte del lado del servidor de una aplicación web y hace referencia a la parte de una aplicación que se encarga del almacenamiento, gestión y recuperación de datos, en pocas palabras de la lógica del sistema y se comunica con la parte del lado del cliente (frontend) de la aplicación para entregar la información deseada al usuario, este fue desarrollado usando el lenguaje de programación Python más exactamente el framework de trabajo Django el cual es muy versátil, poderoso y respeta el patrón de diseño conocido como modelo–vista–controlador.

Además de esto el software cuenta con un asistente inteligente al cual se le podrán realizar preguntas en lenguaje natural, y en función de esto es capaz de entender el contenido de la o las

tablas obtenidas y tratadas y así encontrar la respuesta. El algoritmo es una combinación de dos elementos el primer elemento es la capacidad de entender e inferir de cierta manera cual es el dato que se está preguntando o buscando y la parte del lenguaje natural el cual mediante un preprocesamiento entiende lo que se le está tratando de preguntar, este recibe un input el cual puede ser cualquier pregunta referente al contenido de las tablas eso si por ahora solo se le pueden pasar preguntas en inglés dado que el algoritmo de inteligencia artificial usado se encuentra en una versión beta en desarrollo y funciona manera más precisa en inglés, lo que limita el alcance de consultas pero que a futuro se podrá implementar en español a medida que este algoritmo entre a una etapa más madura. Con el fin de obtener respuestas más rápidas, y dado que la capacidad de procesamiento con la que se cuenta es limitada, es necesario que en el input de la pregunta lleve consigo el año el cual desea ser consultado y para esto se es necesario conocer la estructura de las tablas consultadas.

El Frontend que es la parte del lado del cliente de una aplicación web y es la responsable de la interfaz de usuario y se encarga de la presentación visual de la información al usuario. El frontend interactúa con el backend a través de solicitudes HTTP y recibir respuestas para mostrar información actualizada, relevante al usuario de forma simple y ordenada sin cansar al usuario de la aplicación. Las tecnologías utilizadas en el frontend de nuestro proyecto incluyen HTML, CSS, PHP y JavaScript.

El o los scrappers son un software, programa o script que se utiliza para extraer datos de páginas web el cual funciona automatizando la recolección de información a través de la exploración y análisis de la estructura de un sitio web. Existen muchos framework que se encargan de realizar el mapeo de los sitios web, para este proyecto fueron usados y tenidos en cuenta los

más famosos en la industria como lo son Scrapy, Selenium y beautiful soup ya que cuentan con ciertas características muy interesantes como lo son:

- Automatización: los scrappers automatizan el proceso de extracción de datos, ahorrando tiempo y esfuerzo en comparación con la recopilación manual de información.
- Velocidad: los scrappers pueden recopilar grandes cantidades de datos en un corto período de tiempo.
- Flexibilidad: los scrappers pueden ser configurados para extraer diferentes tipos de datos y personalizados para cumplir con requisitos específicos.
- Integración: los scrappers pueden ser integrados con otras herramientas y sistemas para procesar y analizar los datos extraídos.
- Escalabilidad: los scrappers pueden manejar grandes cantidades de datos y pueden ser escalados según sea necesario.
- Precisión: los scrappers pueden ser programados para asegurar la precisión y consistencia de los datos extraídos.

Tipos de Usuarios:

Tabla 1. *Tipos de Usuarios del Sistema.*

Usuario	Administrador
Tipo de Usuario	Primario
Características	Este puede ser un profesor, directivo o profesional de la institución, encargado de administrar el portal web. Dentro de las funciones de este se encuentra la de registrar a los usuarios, editar los perfiles de usuario y eliminarlos. También este podrá hacer uso y acceder a toda la información presente en el aplicativo web, del mismo modo hacer uso del módulo de inteligencia artificial presente.
Usuario	Usuario autenticado
Tipo de Usuario	Primario
Características	Este puede ser cualquier persona de la institución encargada de realizar el proceso de acreditación de las escuelas que esté interesada en los datos contenido y funcionalidades mostradas en el aplicativo web.

Nota. Tipos de usuarios que están presentes en el sistema con sus determinadas características.

8.2 Requisitos Frecuentes de las Interfaces

- **Interfaz del usuario:** La interfaz será fácil de usar para todos los usuarios, para que puedan aprender a utilizarla de forma rápida. El objetivo es que los usuarios puedan acceder fácilmente a los contenidos, interactuar eficazmente con todos los componentes y sentirse cómodos en forma permanente. Esto se logrará mediante el uso de barras de menús, botones, listas, formularios, imágenes y sliders. La interfaz será diseñada específicamente para el sistema propuesto anteriormente y será visible desde un navegador de internet en cualquier dispositivo (smartphone, Tablet, PC, entre otros).

- **Interfaz del hardware:** Para garantizar un funcionamiento óptimo, es esencial que la institución cuente con computadoras en buen estado que tengan acceso a Internet y los accesorios necesarios. También es importante que el servidor donde se aloja la página cuente con un servicio de soporte que esté disponible las 24 horas del día, los 7 días de la semana.

- **Interfaz del software:** La interfaz está desarrollada utilizando lenguajes de programación como HTML, PHP, JavaScript, Bootstrap y hojas de estilos CSS. Estas tecnologías fueron elegidas para que la interfaz sea fácilmente modificable y adaptable, y además sea compatible con diferentes plataformas y dispositivos.

- **Interfaces de comunicación:** La comunicación entre los servidores, clientes y aplicaciones se realizará a través de protocolos estándar de internet. Por ejemplo, para enviar o recibir archivos o documentos en formato JSON se utilizarán protocolos existentes, como HTTP u otros adecuados.

- **Sistema Operativo:** Windows 10 o superior o Linux (cualquier versión).
Explorador: De preferencia usar Google Chrome o Mozilla Firefox.

8.3 Requisitos Frecuentes de las Interfaces

Los requisitos funcionales pueden ser: cálculos, detalles técnicos, manipulación de datos y otras funcionalidades específicas que se supone, un sistema debe cumplir, en nuestro caso, son los siguientes:

- Es importante que el sistema cuente con un superusuario o administrador que se encargara de llevar el control de usuarios del sistema.
- El sistema le permitirá al administrador acceder y manejar toda la información y datos del portal de manera fácil y automática.
- El sistema debe permitir al usuario elegir entre las diferentes paginas scrapeadas y de esta manera se pueda acceder a toda la información.
- El software deberá permitir al usuario acceder al apartado de gráficos para diseñar sus propios reportes
- El sistema da acceso a una funcionalidad importante de inteligencia artificial a la cual se le deberán realizar preguntas en ingles sobre determinadas tablas.

8.4 Requisitos No Funcionales

En lugar de evaluar el comportamiento específico de un sistema, se establecen criterios que se utilizan para evaluar su operación. Estos criterios se refieren a los requisitos que no describen la información que se debe almacenar ni las funciones específicas que se deben realizar, sino las características de funcionamiento del sistema. En nuestro caso, los criterios específicos son los siguientes:

- El sistema debe asegurar que solo los superusuarios autenticados tengan acceso a la administración y manipulación de la información del sistema mediante la verificación de la identidad de los usuarios.

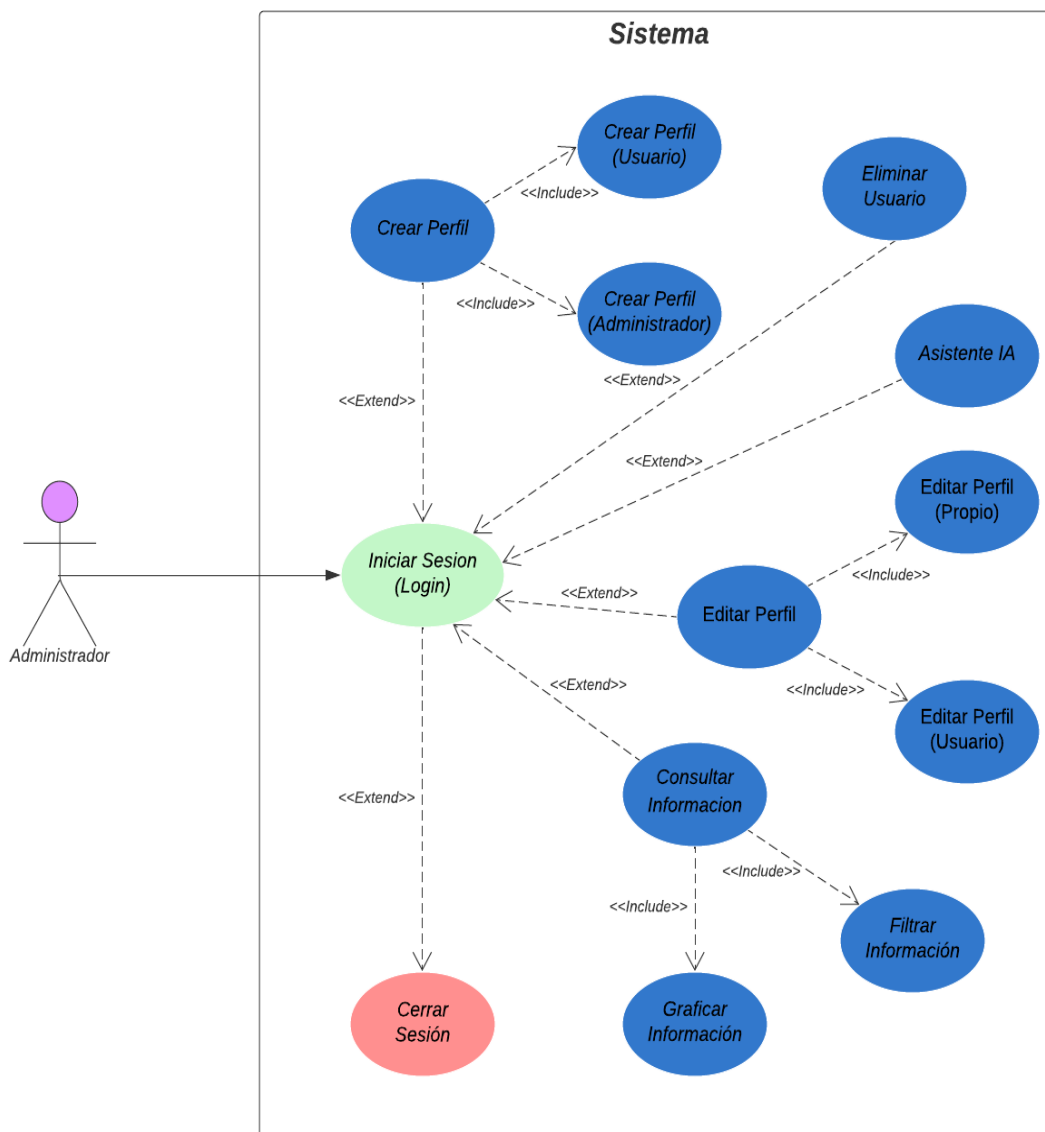
- El sistema debe proporcionar acceso seguro y autenticación de usuarios, además debe ser compatible con cualquier navegador y mostrarse y funcionar correctamente en especial en los navegadores Mozilla y Google Chrome comúnmente usados.
- El sistema debe cumplir con las normas establecidas en la Ley Orgánica de Datos Personales y en el Reglamento de medidas de seguridad, y debe mostrar los resultados de una búsqueda en un plazo no mayor a determinados segundos. Si se excede este tiempo, la búsqueda se detiene y se presentan los resultados obtenidos hasta ese momento.
- El sistema debe contar con una interfaz amigable, fácil de usar y entender, para garantizar la usabilidad en el tiempo. La interfaz debe ser atractiva y sencilla, de modo que el usuario se sienta cómodo al utilizarlo y pueda sacar el máximo provecho de las funciones que ofrece, para facilitar y agilizar su trabajo.

8.5 Diagramación UML

Partiendo de la especificación de requisitos funcionales y la descripción de los usuarios del sistema (administrador y usuario autenticado) y de los casos de uso y de todo el análisis anterior se realizaron los siguientes diagramas UML de casos de uso.

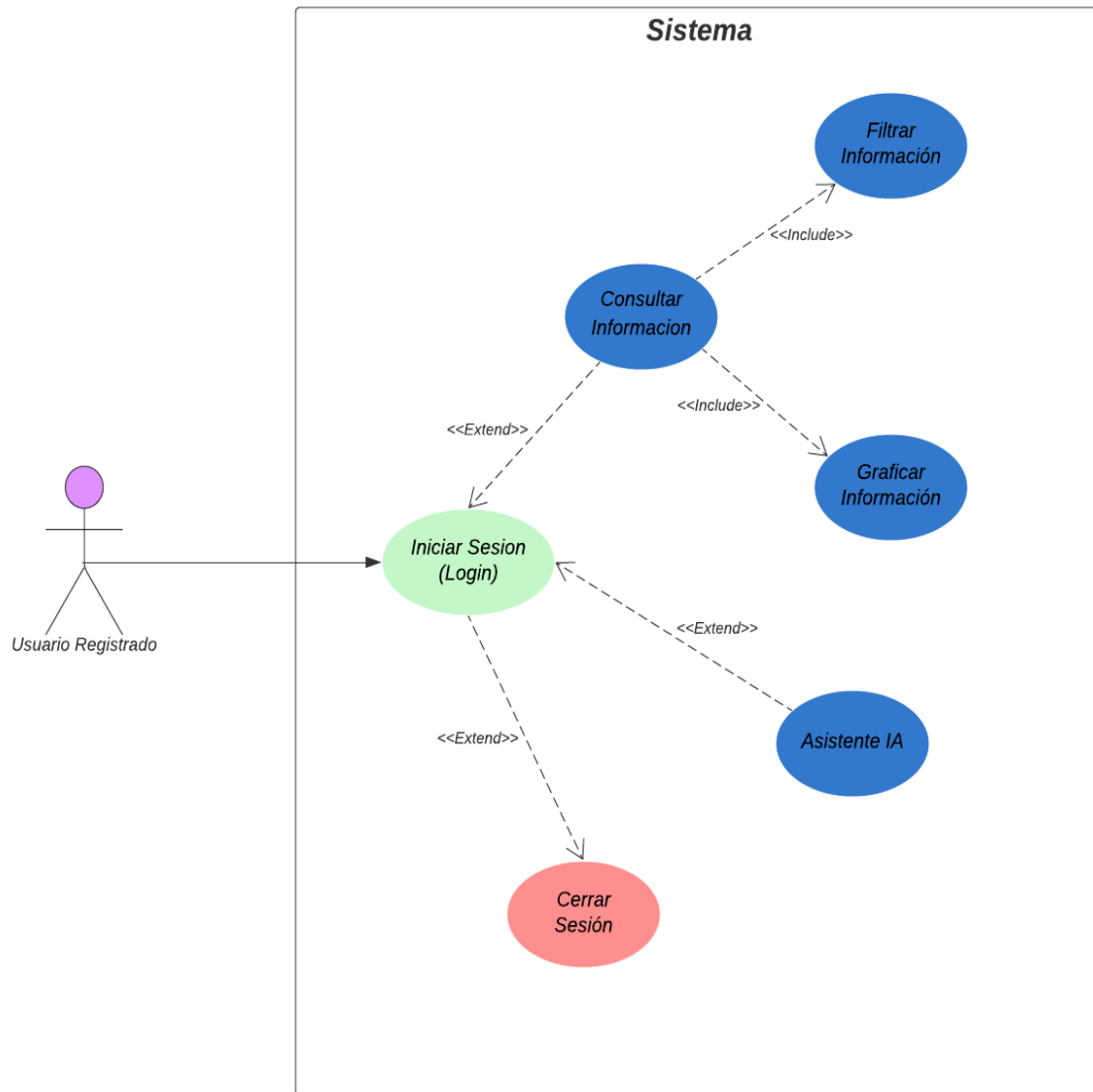
8.5.1 Diagramas de Casos de Uso

Figura 3. Diagrama Casos de Uso para el Administrador.



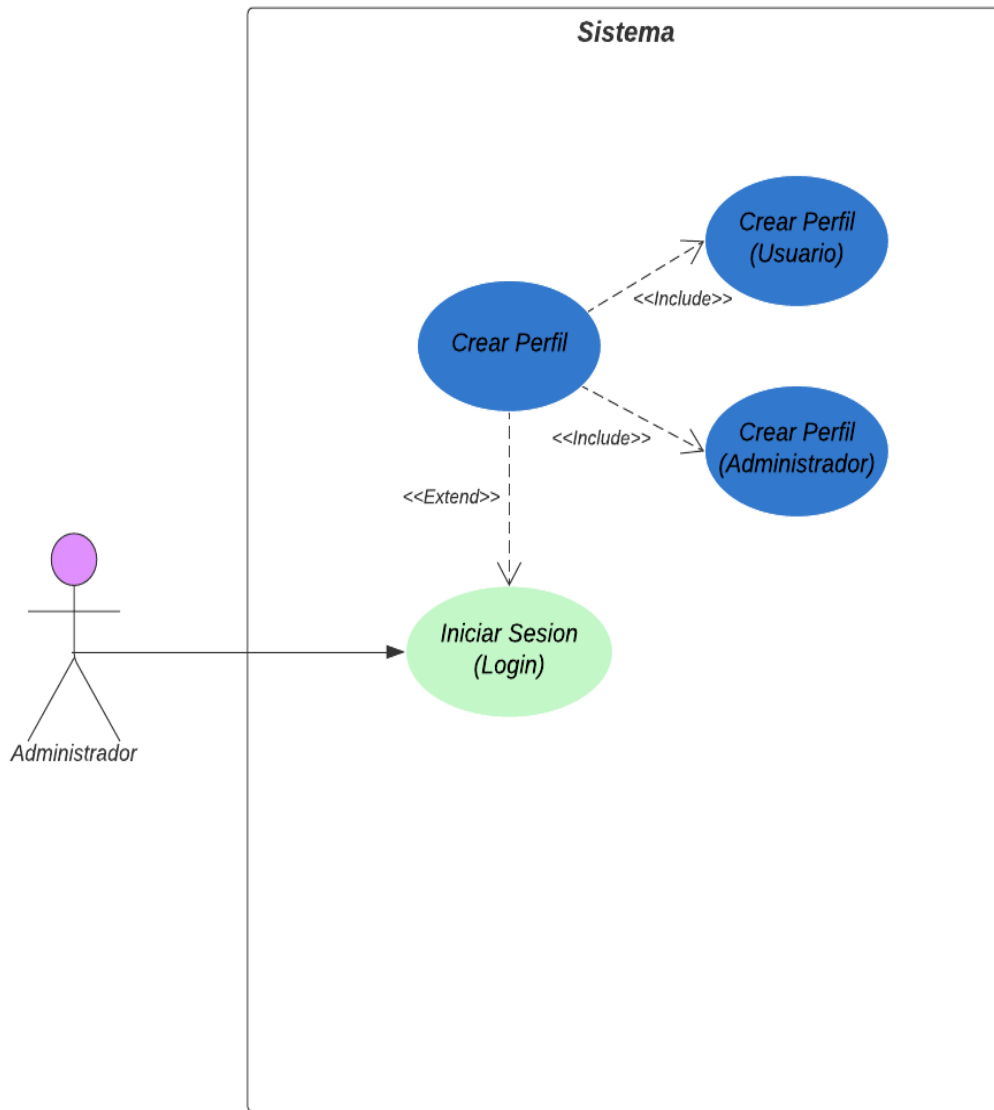
Nota. El grafico muestra el caso de uso general para el rol de administrador.

Figura 4. Diagrama de Casos de Uso para el Usuario Autenticado.



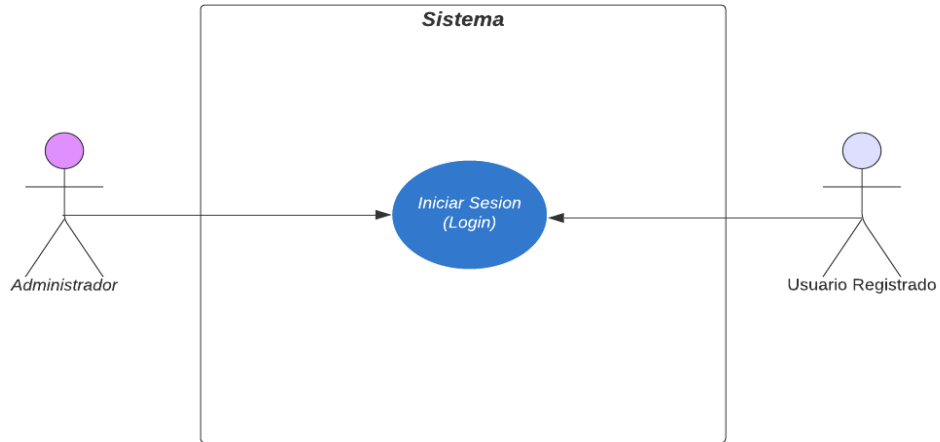
Nota. El gráfico muestra el caso de uso para el usuario registrado en el sistema.

Figura 5. Diagrama de Casos de Uso Registrar Usuarios



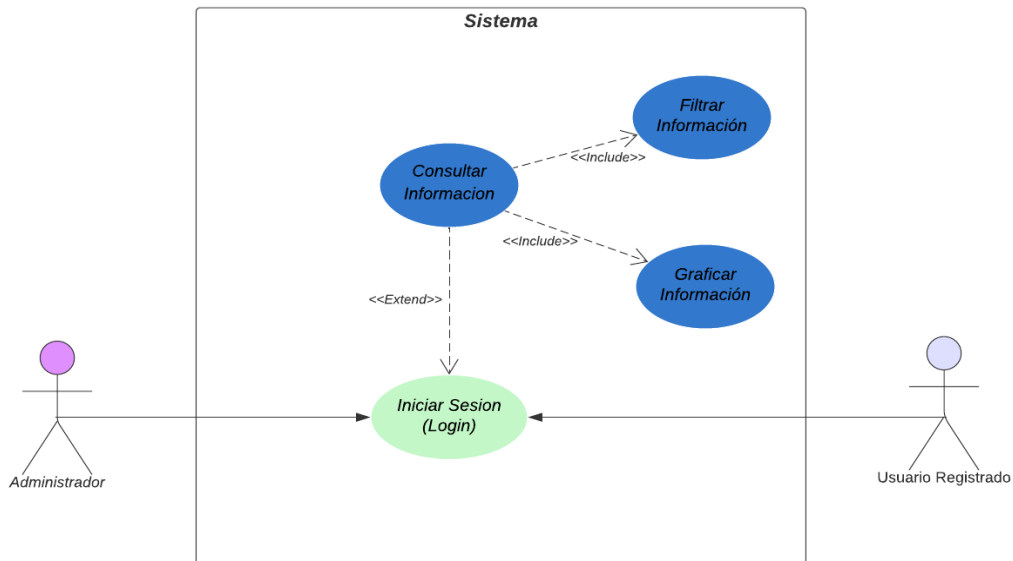
Nota. El grafico muestra el caso de uso registrar usuarios para el rol del administrador.

Figura 6. Diagrama de Casos de Uso Iniciar Sesión.



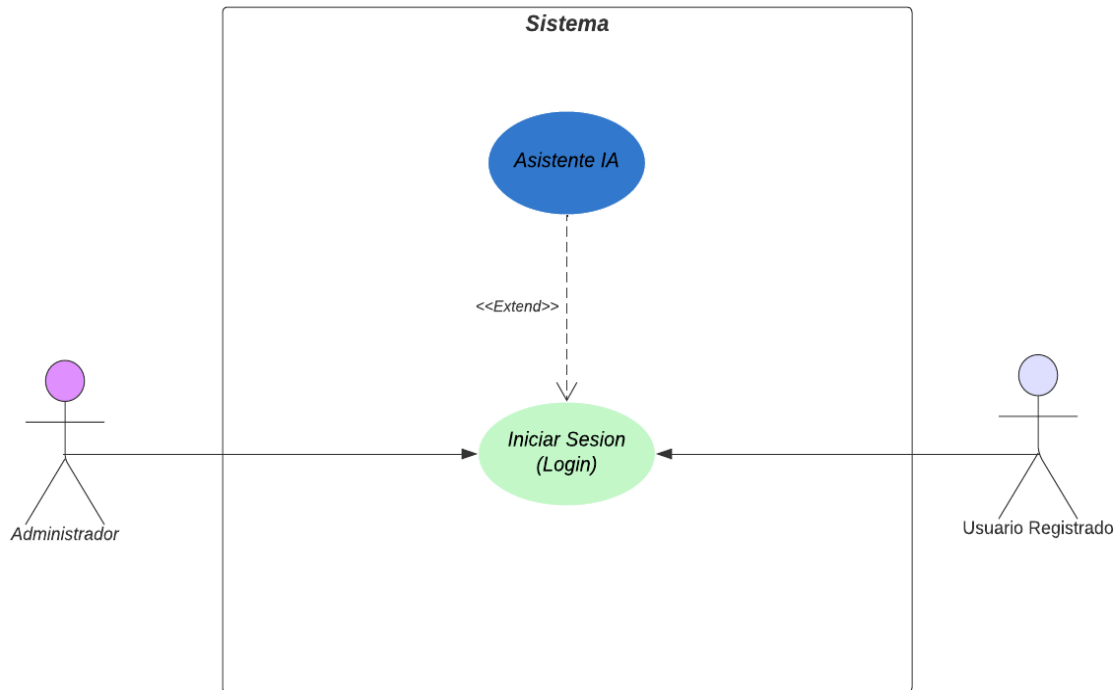
Nota. El grafico muestra el caso de uso iniciar sesión para el rol del administrador.

Figura 7. Diagrama de Casos de Uso Consultar, Filtrar y Graficar Información.



Nota. El grafico muestra el caso de uso para la función graficar y filtrar información.

Figura 8. Diagrama de Casos de Uso Consultar Asistente de IA.



Nota. El grafico muestra el caso de uso para la inteligencia artificial.

8.6 Diseño

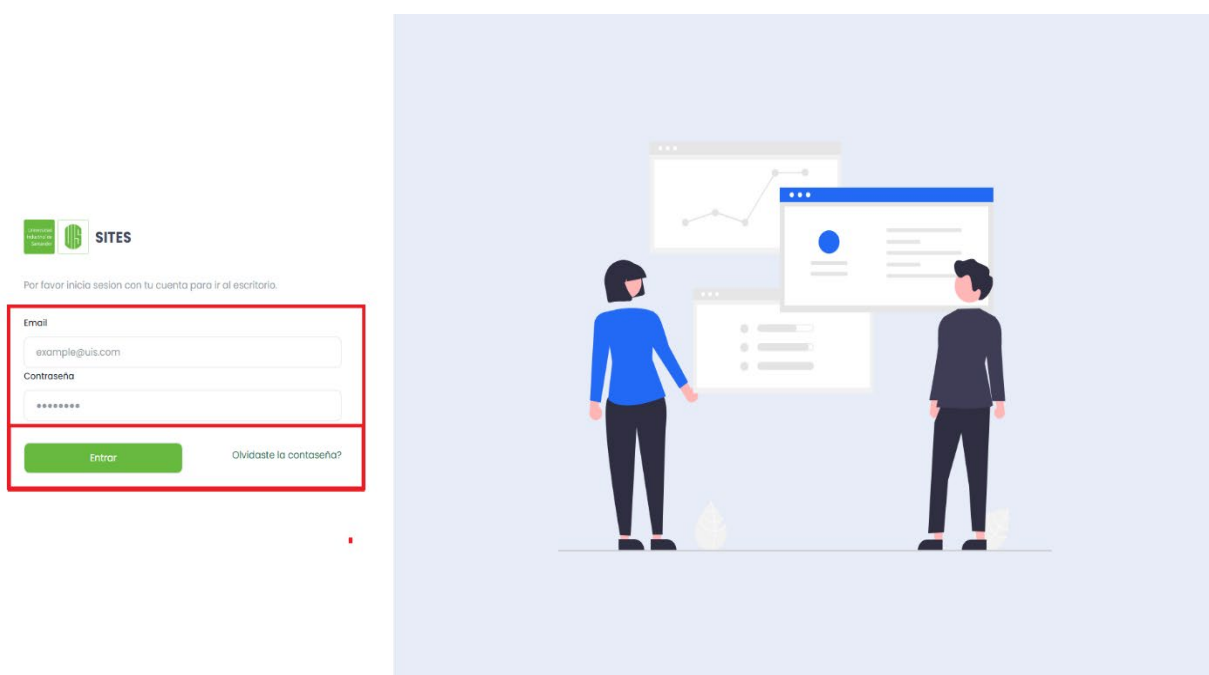
Para el diseño mostramos las vistas del usuario con su respectiva explicación de botones y funcionamiento detallado para que los usuarios del software puedan aprender a usarlo de forma fácil.

8.6.1 Manual de Usuario

A continuación, se explicarán a detalle cómo realizar cada acción disponible dentro de las diferentes funciones y opciones a realizar que brinda el sistema

Acceso al Sistema: La pantalla del sistema muestra el módulo de login, en la cual en el campo “Email” se debe colocar el nombre de usuario, en el campo “Contraseña” su respectiva contraseña y hacer clic en “Entrar” para obtener su respectivo acceso, en caso contrario que haya olvidado su contraseña puede hacer clic en el botón “Olvidaste tu contraseña” para poder generar una nueva.

Figura 9. Inicio de sesión del administrador y usuarios inscritos en el sistema.



Nota. El grafico muestra la pantalla para el inicio de sesión tanto del administrador como de los usuarios.

Si la combinación de nombre de usuario y contraseña es correcta, se permitirá el acceso al sistema. De lo contrario, se requerirá que se vuelvan a ingresar ambos datos.

Una vez se accede correctamente se le mostrara el escritorio del aplicativo, el cual en la parte izquierda de la imagen se observan el menú el cual está compuesto por 5 ítems llamados: “inicio”, “Carreras”, “Usuarios”, “Asistente Inteligente” y “Documentación”.

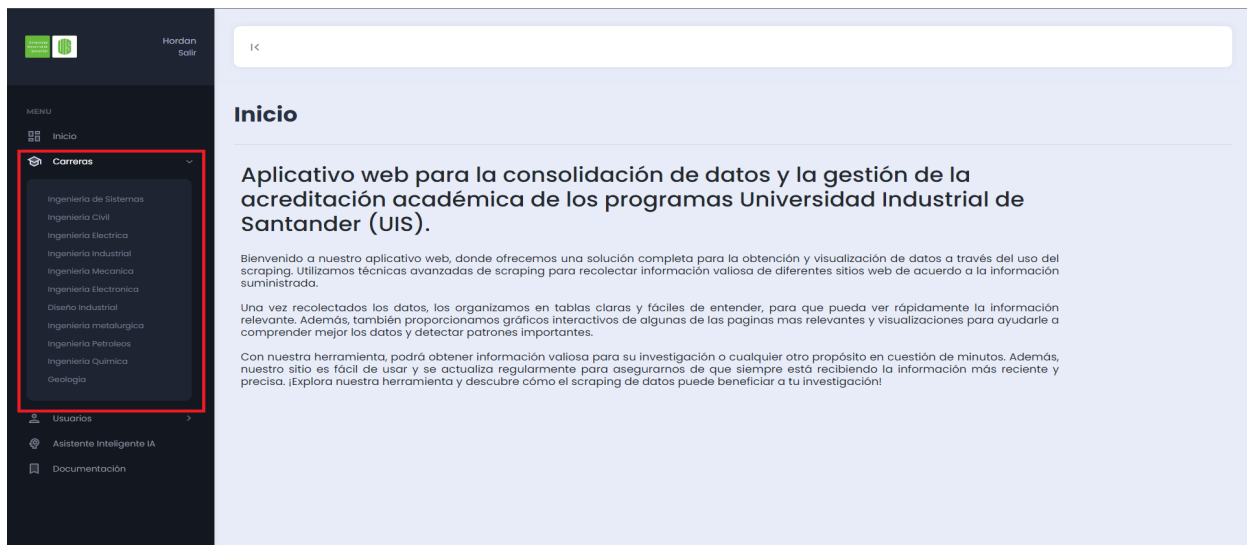
Figura 10. *Página principal o escritorio del perfil administrador de la aplicación.*



Nota. El grafico muestra la pantalla de inicio del aplicativo luego de iniciar sesión.

A continuación, se despliega el menú “Carreras” el cual está compuesto de todas carreras de las facultades de ingeniería fisicoquímicas y físico-mecánicas de la Universidad Industrial de Santander.

Figura 11. *Página principal del perfil administrador de la aplicación con el menú “Carreras” desplegado.*



Nota. El grafico muestra las carreras luego de ser desplegado el menú “Carreras”.

Gestión de usuarios: Para la gestión de usuarios, el administrador de la aplicación puede ingresar y modificar información de los usuarios que van a tener acceso a la aplicación para el cual debe ir a la pestaña “Usuarios” como se muestra a continuación.

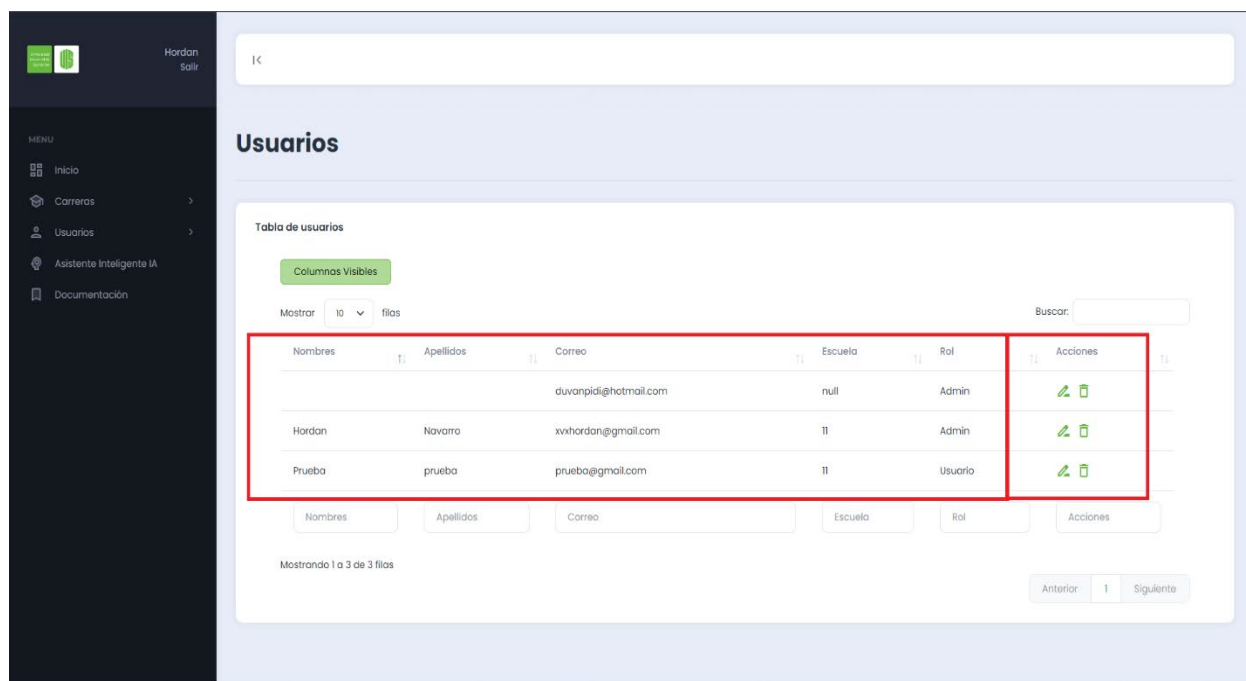
Figura 12. *Página principal del perfil administrador de la aplicación con el menú “Usuarios” desplegado.*



Nota. El grafico muestra el menú desplegado de “Usuarios” para el perfil del administrador.

Una vez seleccionada la opción “Usuarios” se muestra la pestaña para la gestión de los usuarios por parte del administrador en el cual se observan los perfiles de usuario y las acciones que puede realizar el administrador con los diferentes perfiles del aplicativo.

Figura 13. Menú “Gestionar Usuarios”.



Nota. El grafico muestra los perfiles registrados en el sistema para su respectiva gestión.

Para crear un perfil el administrador deberá seleccionar en el menú desplegable de “Usuarios” la opción de “Añadir nuevo” el cual le mostrará los campos requeridos para la creación de un nuevo perfil, se deberán llenar cada uno de los campos sin excepción y además este cuenta con un botón el cual autogenera una contraseña por si el usuario así lo desea. Una vez hecho esto se deberá dar clic en el botón “Añadir”

Figura 14. Menú “Anadir Usuario”.

The screenshot shows the 'Añadir usuario' form. It features a dark sidebar with a menu containing 'Inicio', 'Carreros', 'Usuarios', 'Asistente Inteligente IA', and 'Documentación'. The main content area has a search bar and a form with the following fields: 'Nombres' and 'Apellidos' (text inputs), 'Carreo' (text input), 'Numero de contacto' and 'Documento' (text inputs), 'Escuela' (dropdown menu with 'Escoger...' selected), and 'Rol' (dropdown menu with 'Escoger...' selected). There is a 'Contraseña' field (text input). Two green buttons are present: 'Añadir' at the bottom left and 'Generar' at the bottom right. A red box highlights the entire form area.

Nota. El grafico muestra el formulario a llenar para registrar un usuario en la plataforma.

Para actualizar un usuario el administrador debe dar clic sobre el botón ”Editar” situado al lado derecho del usuario, a continuación, mostrándole el formulario de edición de los perfiles.

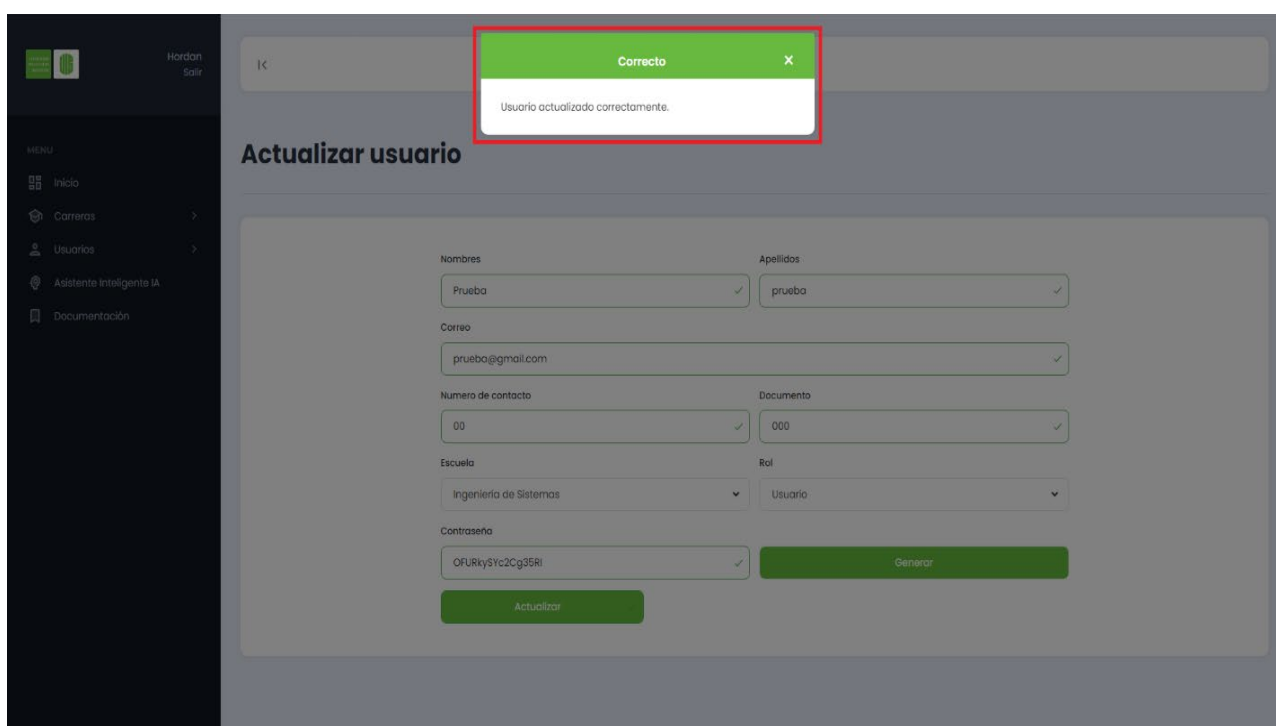
Figura 15. Formulario Editar o Actualizar Usuario.

The screenshot shows the 'Actualizar usuario' form. It features the same dark sidebar as Figure 14. The main content area has a search bar and a form with the following fields: 'Nombres' (text input with 'Prueba' entered), 'Apellidos' (text input with 'prueba' entered), 'Carreo' (text input with 'prueba@gmail.com' entered), 'Numero de contacto' and 'Documento' (text inputs with '00' entered), 'Escuela' (dropdown menu with 'Ingeniería de Sistemas' selected), and 'Rol' (dropdown menu with 'Usuario' selected). There is a 'Contraseña' field (text input). Two green buttons are present: 'Actualizar' at the bottom left and 'Generar' at the bottom right. A red box highlights the entire form area.

Nota. El grafico muestra el formulario para actualizar o editar a los usuarios registrados en el sistema.

Una vez llenos los espacio a editar se procede a dar clic en el botón “Actualizar”, si el proceso es exitoso se mostrará un mensaje que dice “Usuario actualizado correctamente” en caso contrario le pedirá que repita el proceso llenando los espacios requeridos.

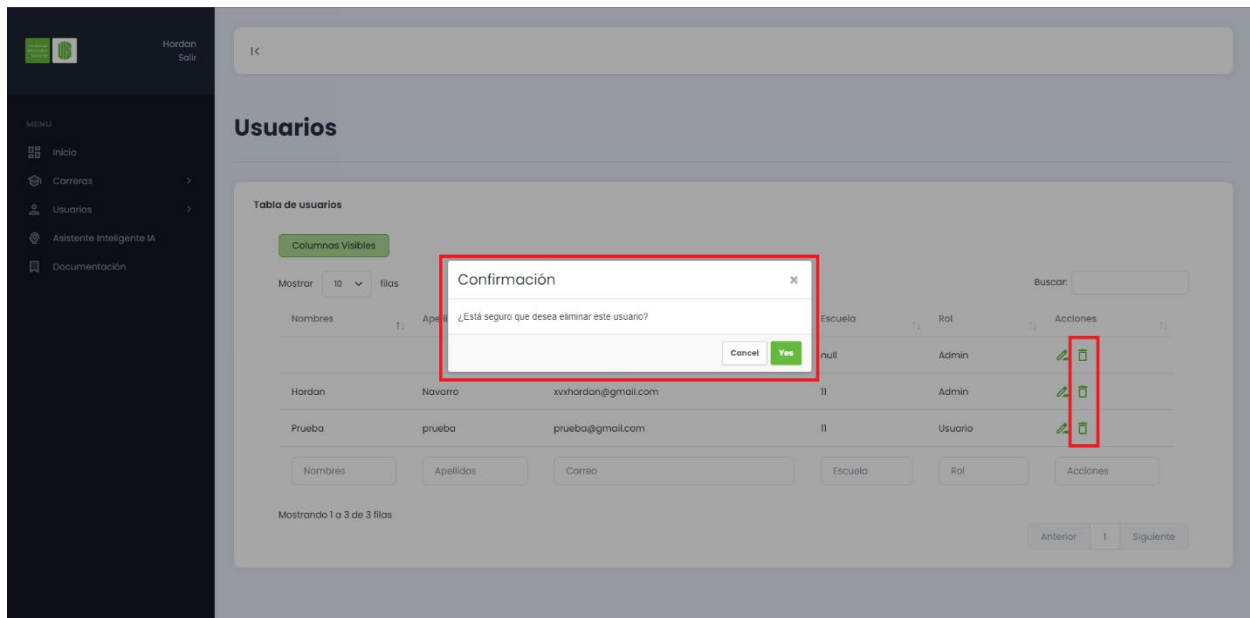
Figura 16. *Usuario editado o actualizado correctamente.*



Nota. El grafico muestra el mensaje de “Correcto” cuando el usuario ha sido editado con éxito.

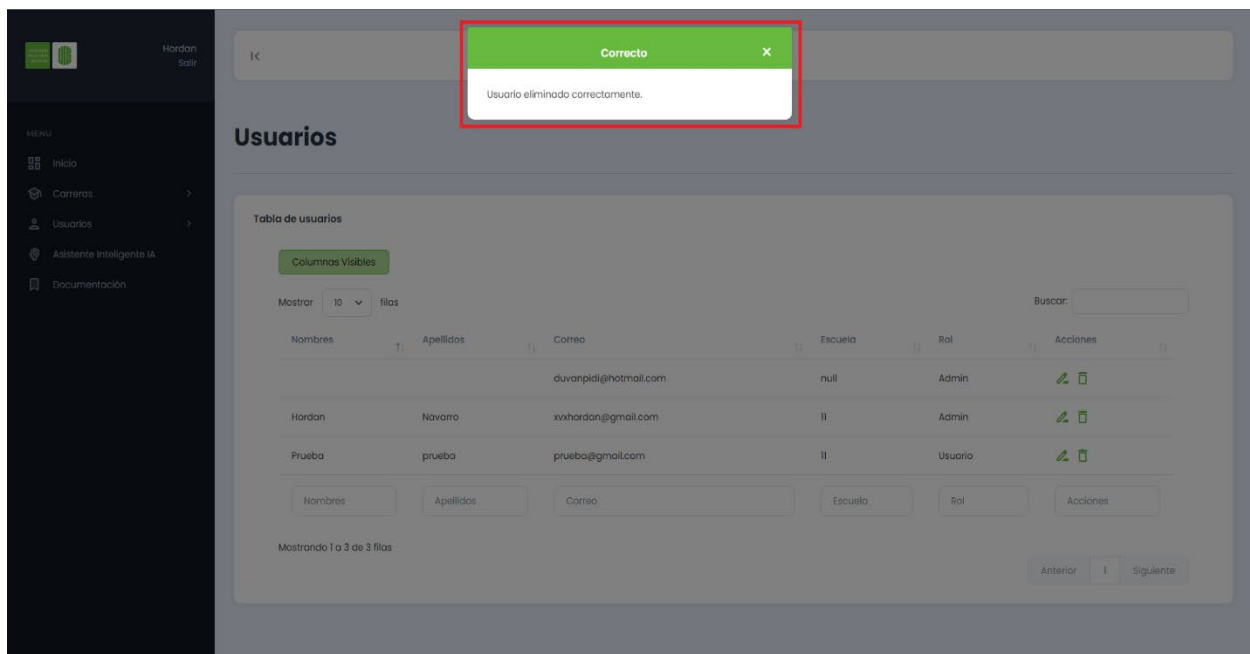
Para eliminar un usuario el administrador debe dar clic sobre el botón “Eliminar” situado en la parte derecha.

Figura 17. Eliminar usuario.



Nota. El grafico muestra el botón “Eliminar usuario” y el mensaje de confirmación antes de eliminarlo.

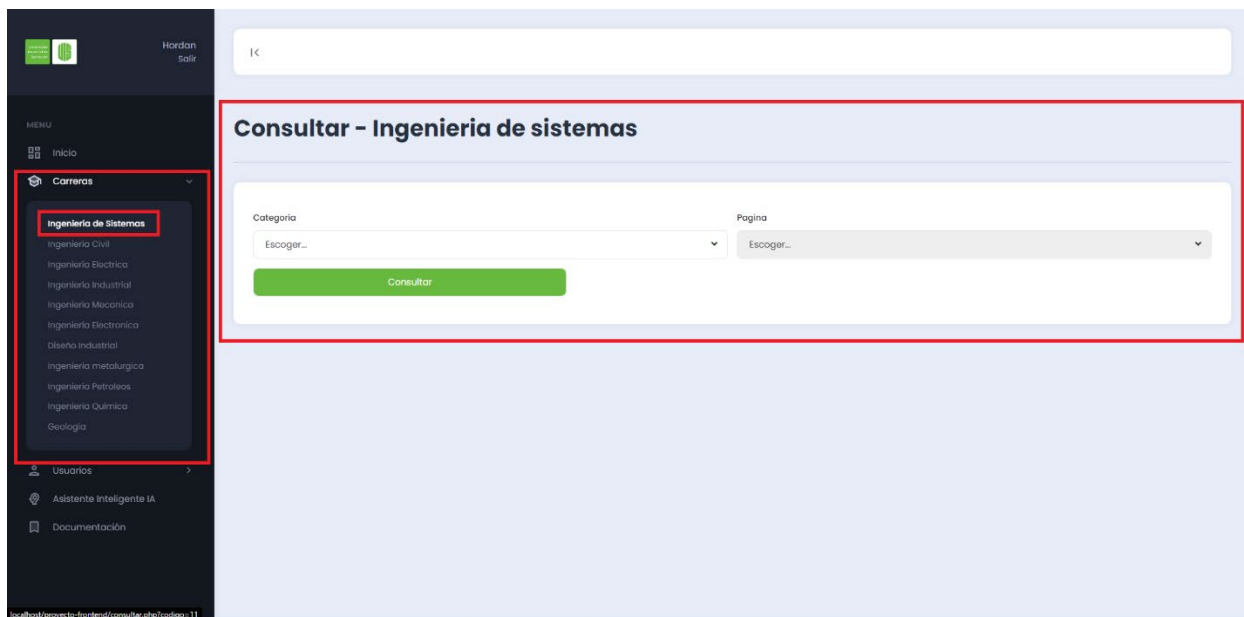
Figura 18. Mensaje confirmación eliminar usuario.



Nota. El grafico muestra el mensaje “Correcto” luego haber eliminado satisfactoriamente a un usuario registrado.

Consultar Información: Para consultar la respectiva información es necesario ubicarse en el menú principal situado a la izquierda de la pantalla y seleccionar el menú desplegable “Carreras” y seleccionar la carrera de la cual desea obtener información para posteriormente mostrar la vista de “Consultar”.

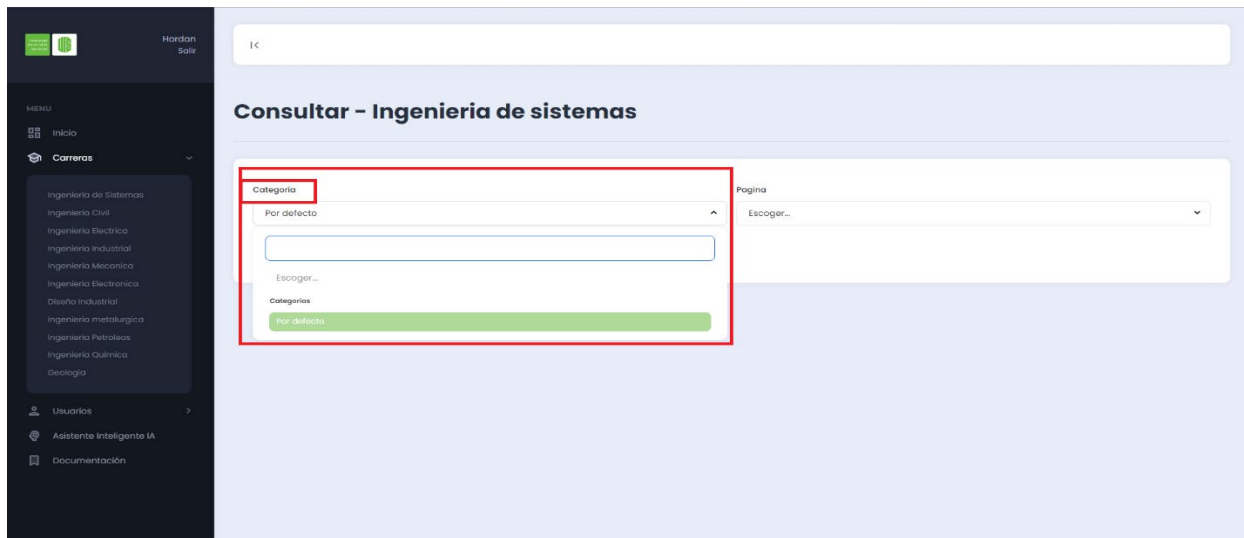
Figura 19. *Seleccionar carrera a consultar.*



Nota. El grafico muestra el escritorio de la sección “Consultar” luego de seleccionar una determinada carrera.

Ya adentro de este menú se muestran dos menús desplegables uno de ellos llamado “Categoría” la cual por ahora solo hay una creada llamada “Por defecto” y se debe seleccionar para realizar la búsqueda.

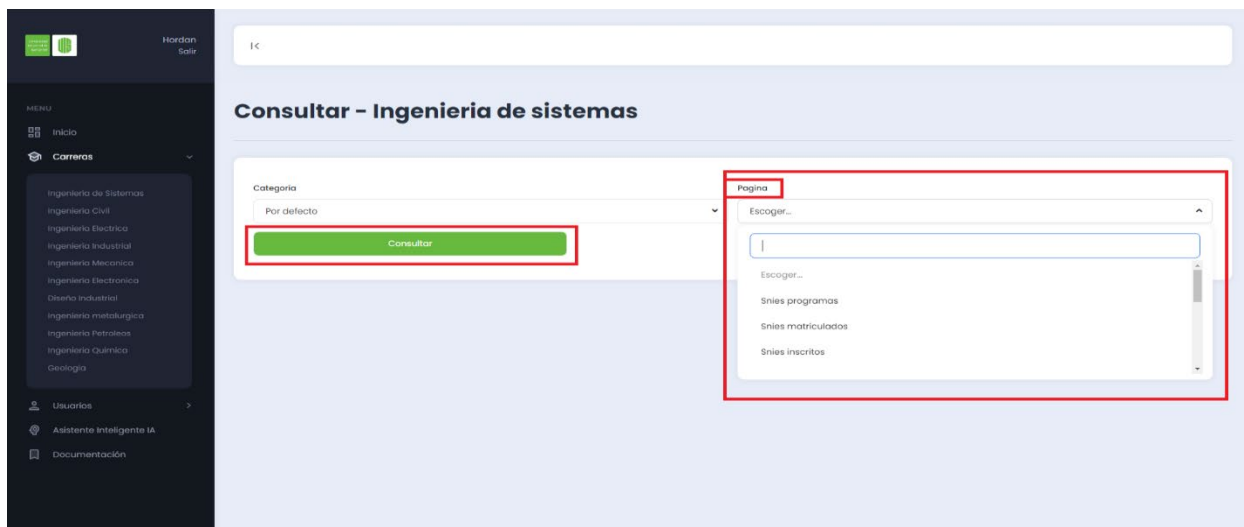
Figura 20. *Seleccionar categoría.*



Nota. El grafico muestra el menú “Categoría” en el cual están contenidas todas las paginas consultada.

El otro menú se llama “Página” el cual posee toda la información obtenida y tiene una larga lista de posibilidades entre las cuales se encuentran el SNIES, el servicio de empleo, entre otras, a continuación, se selecciona la página de interés y se da clic en consultar.

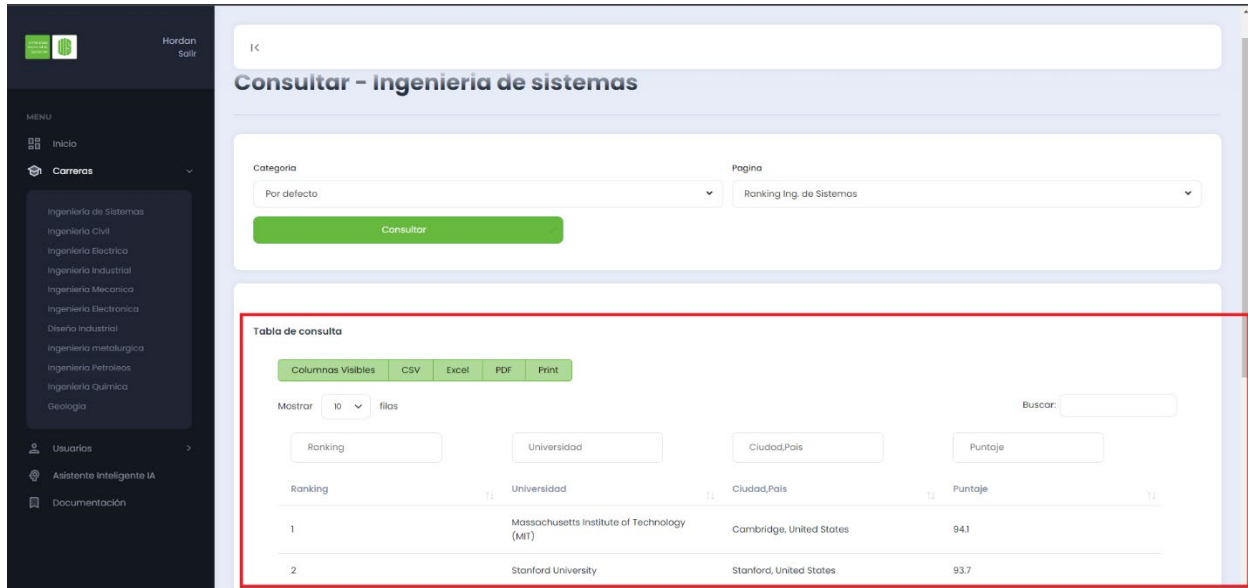
Figura 21. *Seleccionar página.*



Nota. El grafico muestra el menú desplegable “Página” en el cual podrá seleccionar la página que desea consultar.

Una vez consultada la página podemos observar a la información cargar en pantalla.

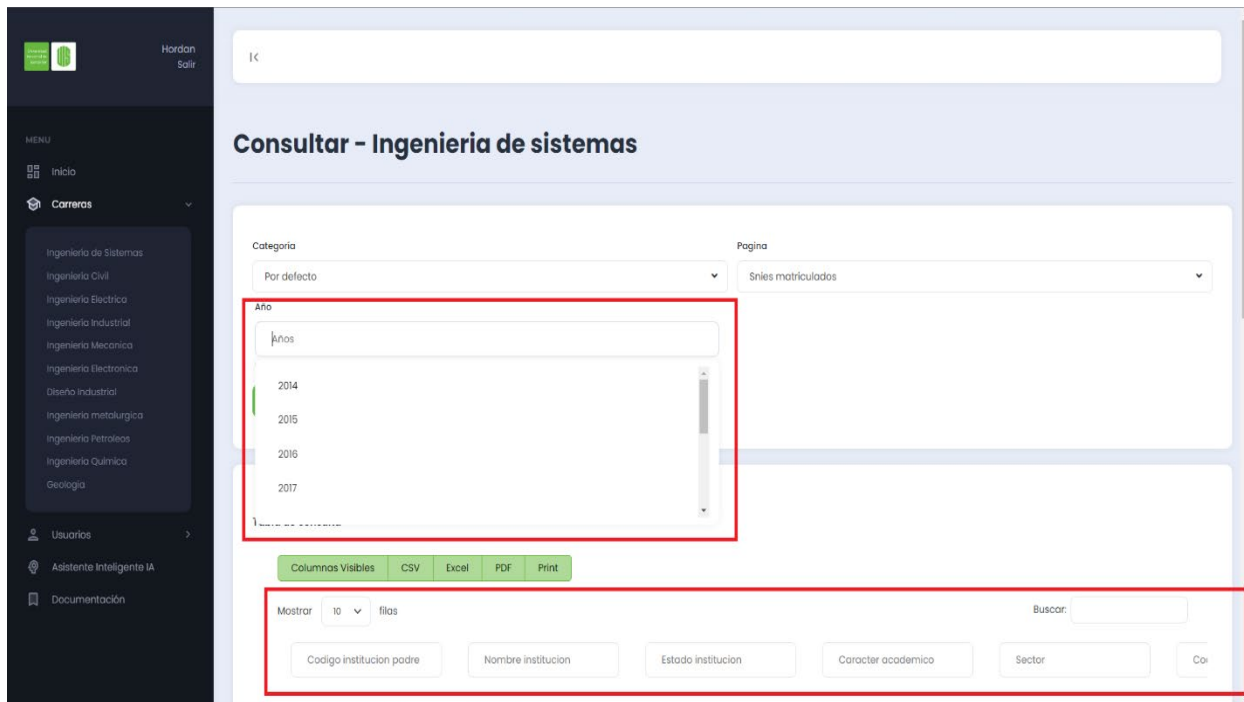
Figura 22. Vista Información consultada.



Nota. El grafico muestra información consultada de alguna página seleccionada.

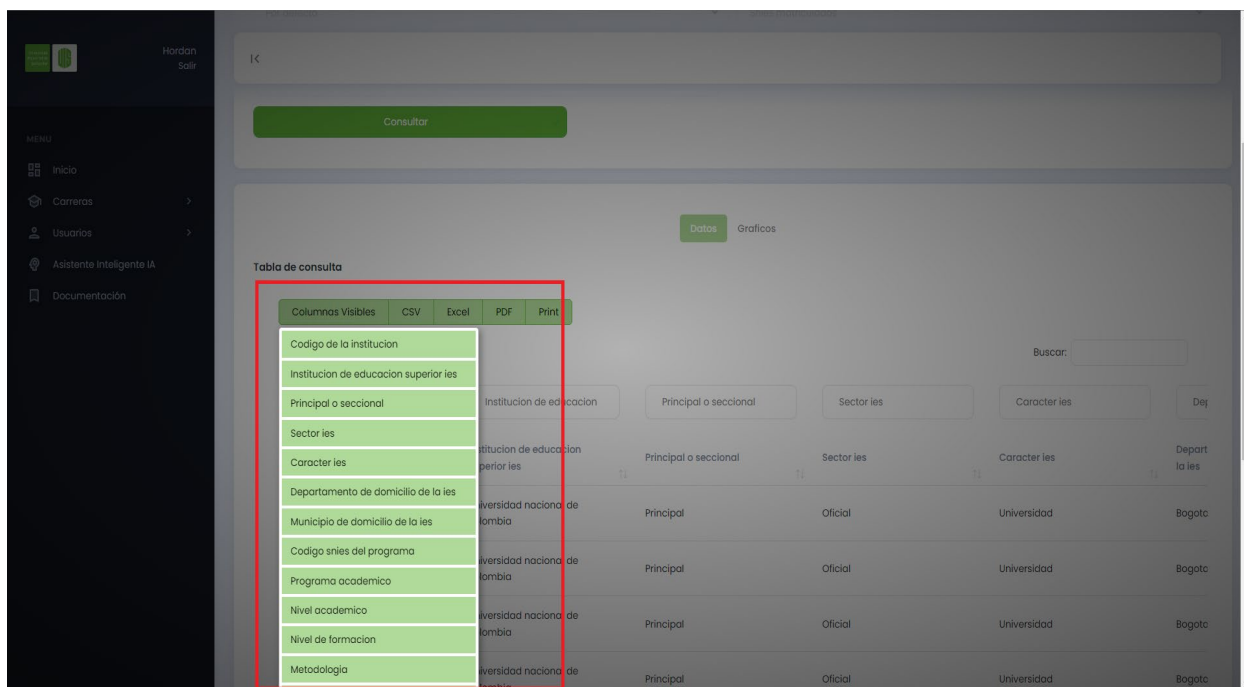
Filtrar Información: Para el filtrado de información se tuvieron en cuenta ciertos parámetros como la cantidad de información, para realizar su respectiva paginación, años, nombre de claustros educativos entre otros. Cada página consultada contiene determinada información y de acuerdo con esta se le aplicaron filtros los cuales se muestran solo la información asociada al filtro aplicado.

Figura 23. Ejemplos de filtros presentes 1.



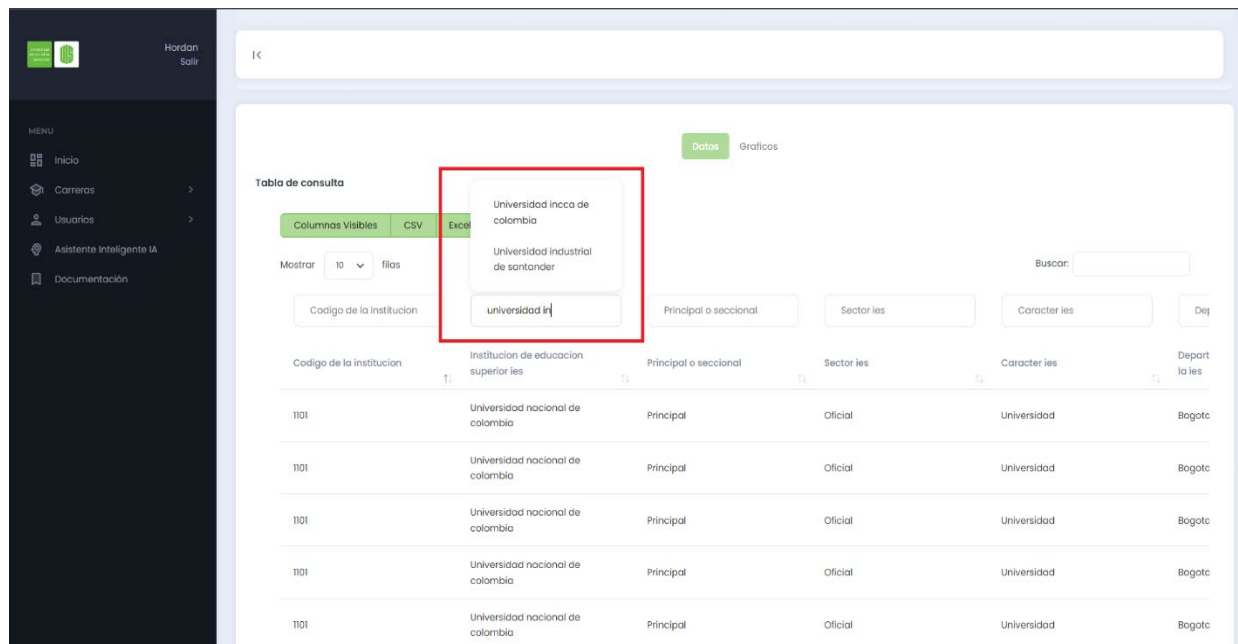
Nota. El grafico muestra filtros presentes para clasificar la información.

Figura 24. Ejemplos de filtros presentes 2.



Nota. El grafico muestra filtros presentes para clasificar la información.

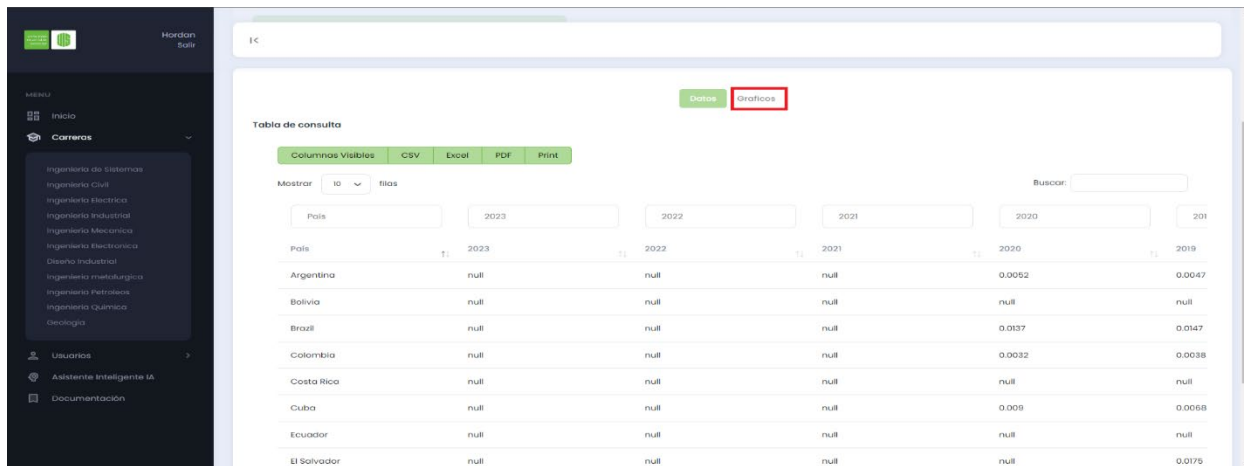
Figura 25. Ejemplos de aplicación de filtros a una página.



Nota. El grafico muestra un ejemplo de cómo aplicar un filtro para clasificar la información.

Graficar Información: Para graficar la información de una página es necesario que la pagina consultada cuente con este atributo dado que no todas poseen este aparatado por el tipo de datos que poseen. Para acceder al este apartado es necesario haber realizado una consulta y encontrarse en esta pestana, estando ahí veras si cuenta con el atributo y lo siguiente es dar clic sobre el botón “Gráficos”.

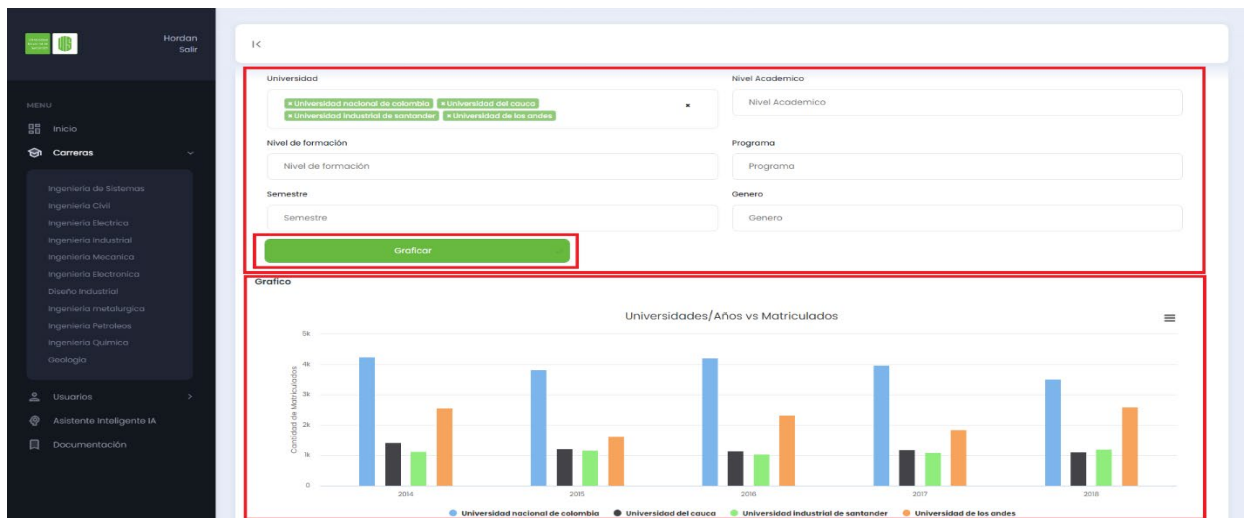
Figura 26. Seleccionar el apartado de “Gráficos”.



Nota. El grafico muestra el botón de la sección “Graficar Información”.

Una vez adentro se mostrarán los diferentes filtros con los que fue diseñado cada espacio para realizar la respectiva grafica de la información. Lo siguiente es seleccionar los respectivos menús desplegables que contienen los diferentes filtros que se desean aplicar para realizar la respectiva ilustración de la información, finalmente se debe dar clic sobre el botón graficar.

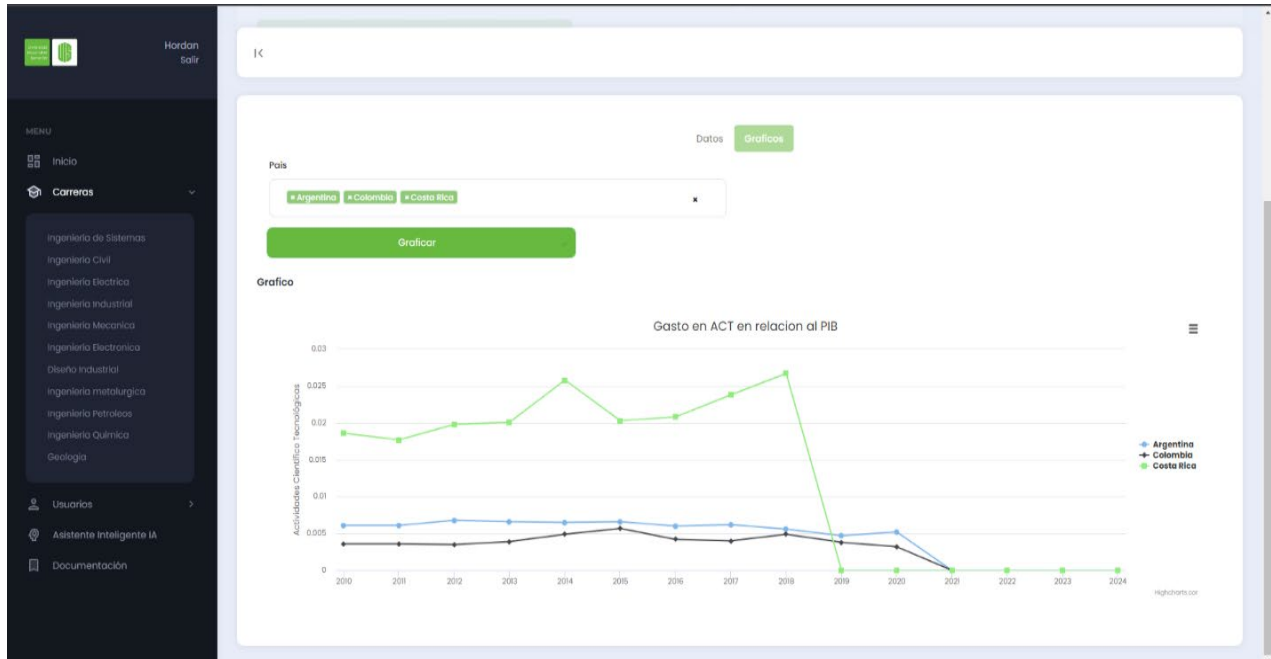
Figura 27. Graficar información 1.



Nota. El grafico muestra un ejemplo de la sección “Graficar Información”.

Como se dijo anteriormente cada página trae un paquete de información específico por lo cual la forma de graficar la información cambia con cada página consultada, a continuación, veremos otro ejemplo de cómo se ilustra la información obtenida.

Figura 28. Graficar información 2.



Nota. El grafico muestra un ejemplo de la sección “Graficar Información”.

Consultar Asistente Inteligente: Para realizar una consulta en el asistente inteligente es necesario dirigirse al menú de la parte izquierda y dar clic sobre el apartado que dice “Asistente inteligente”, a continuación, se muestra la vista donde encontraremos el input donde se tendrá que poner la pregunta en ingles acotando la búsqueda por años.

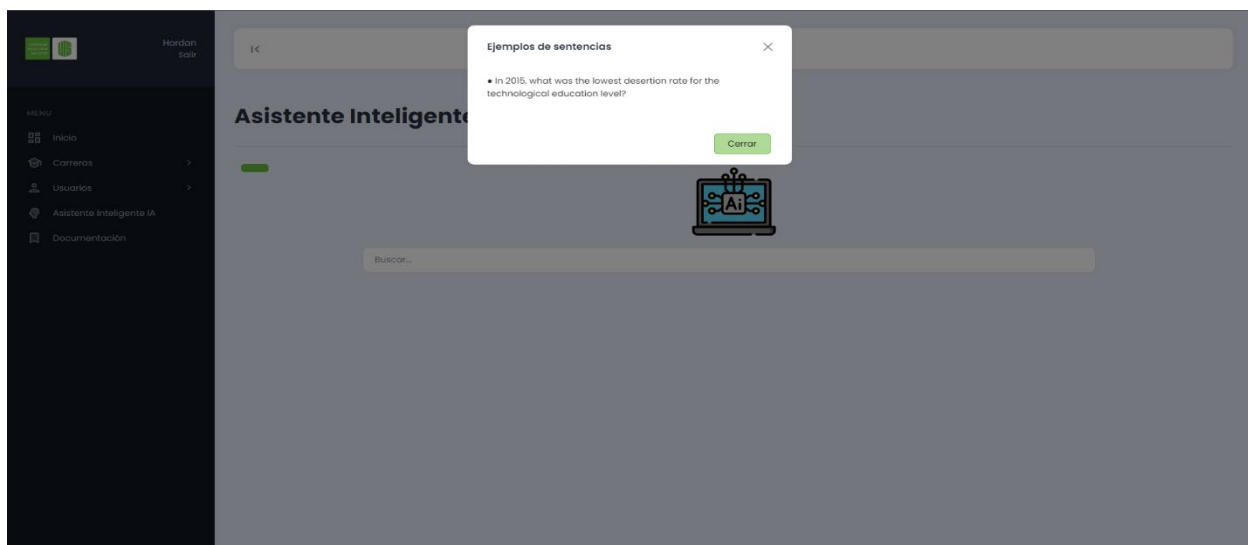
Figura 29. Vista del “Asistente inteligente IA”.



Nota. El grafico muestra el escritorio del “Asistente Inteligente IA” con su respectiva caja de búsqueda.

Una vez allí se debe colocar la pregunta y presionar la tecla “Enter” para luego obtener la determinada respuesta. A continuación, se muestra un ejemplo de algunas sentencias o preguntas que se le pueden realizar al asistente.

Figura 30. Ejemplo de preguntas para el “Asistente inteligente”.



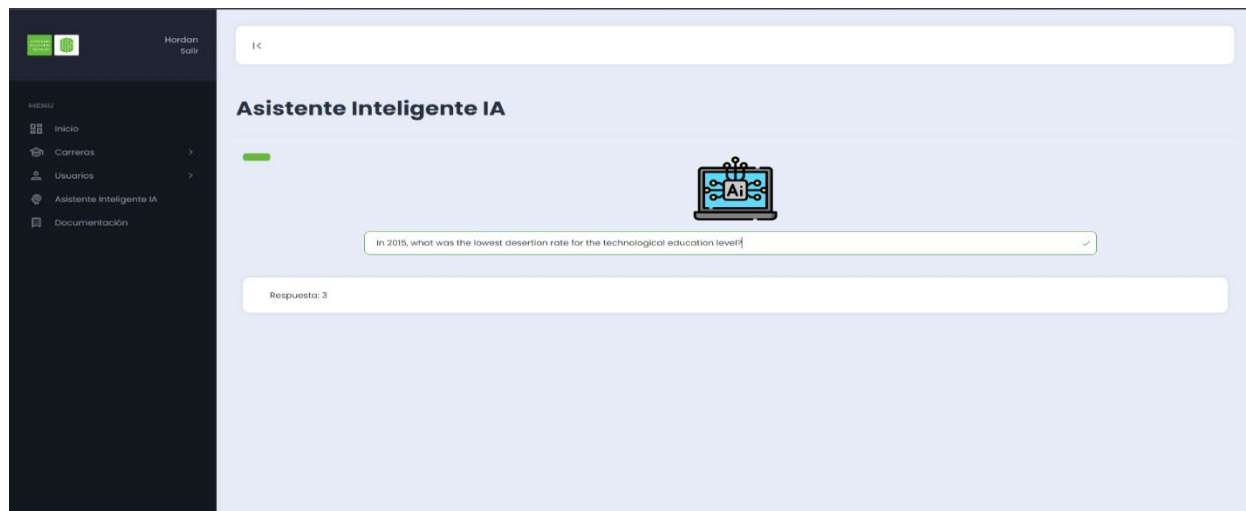
Nota. El grafico muestra algunos ejemplos de preguntas para el “Asistente Inteligente IA”

A continuación, se muestra un ejemplo de una pregunta respondida por el asistente

Pregunta: in 2015, what was the lowest desertion rate for the technological education level?

La respuesta fue: 3

Figura 31. Ejemplo de una consulta al “Asistente inteligente”.



Nota. El grafico muestra un ejemplo concreto de una pregunta realizada a el “Asistente Inteligente IA” con su respectiva respuesta.

Llegados hasta aquí podemos decir que en el desarrollo e implementación de esta aplicación y de la versión final propuesta se presentaron algunos inconvenientes en la claridad y documentación de muchos pasos; se incurrió en redundancias e inconsistencias en la concepción de los procesos, esto debido a la falta o poca experiencia en documentación por parte de los autores, además se presentaron algunos obstáculos al momento de implementar los script que son la lógica de cada algoritmo de scrapper, pues como sabemos cada página web posee su propia lógica y estructura y puede ir cambiando con el tiempo haciendo que estos queden obsoletos en el tiempo solo con que el HTML o estructura de la página presente algunos cambios. También en la utilización de la inteligencia artificial se dejó una base de la aplicación de un algoritmo de red

neuronal el cual aún está en proceso de desarrollo, pues presenta ciertas limitaciones, como lo es el uso del lenguaje a la hora de realizarle una pregunta o que el input con el fin de ahorrar recursos de procesamiento se le deba pasar un año específico el cual desea ser consultado, estos ajustes se realizaron con el fin de brindar un asistente eficaz y funcional en su labor, se recomienda tenerlo en cuenta para futuras versiones pues presenta características útiles y muy poderosas aplicables a este tipo de software en específico y que podrían sacarle el máximo partido a este tipo de proyectos, en general se presenta un software ligero pero poderoso en su labor.

9. Conclusiones

Con el desarrollo de este aplicativo se logró recolectar y centralizar una gran cantidad de información valiosa para el proceso de acreditación en una sola base de datos, disponible para ser consultada en cualquier momento del día, lo que evita a los profesionales encargados de obtenerla tener que ir fuente por fuente recolectando dicha información.

Dada la basta cantidad de información necesaria para realizar el proceso de acreditación institucional y a la poca colaboración que tuvimos por parte de las escuelas beneficiadas con el aplicativo brindándonos las fuentes recolección, la información extraída y presente en el software maneja casi las mismas fuentes para todas las carreras ya que las páginas consultadas manejan información general útil para todas las escuelas en general.

La automatización del aplicativo hará que se mantenga actualizado en el tiempo puesto que fue programado para que cada semana autoejecute los scrappers, descargue los archivos y/o datos necesarios, procese la información y la mantenga al día para su respectivo uso .

El uso de inteligencia artificial en el proyecto a través de la adaptación y acoplamiento de un algoritmo de red neuronal en fase de desarrollo, el cual recibe consultas en lenguaje natural, resulta ambicioso y beneficioso porque establece una base para su uso y aplicación, permitiendo a futuro obtener respuestas basadas en el análisis de la información con la que fue pre-entrenado.

10. Trabajos Futuros

Como trabajo futuro se recomienda hacer una recolección y análisis a profundidad de las fuentes de extracción de la información carrera por carrera con el fin de consolidarla y obtener no solo información de interés general sino también específico para cada escuela.

Se recomienda estar atentos a los scripts desarrollados para automatizarlos, pues como sabemos cada página web posee su propia lógica y estructura, esta puede ir cambiando con el tiempo haciendo que estos scripts queden obsoletos solo con que el HTML o estructura de la página presente algunos cambios.

La precisión de la respuesta obtenida usando el módulo de IA está sujeta a la forma en que el usuario formule la pregunta, adicionalmente estas preguntas deben ser redactadas en idioma inglés ya que el modelo usado fue pre-entrenado en datos tabulares en este idioma. Esta es actualmente una restricción del modelo y se presenta como una oportunidad de mejora para futuras versiones de la aplicación.

Referencias Bibliográficas

- Amazon. (s.f.) ¿Qué es JavaScript?. Consultado el 2 de diciembre de 2022. Amazon. <https://aws.amazon.com/es/what-is/javascript/>
- AUTHENTICATION. (s.f.). Authentication - Django REST framework. Consultado el 28 de noviembre de 2022. <https://www.django-rest-framework.org/api-guide/authentication/>
- Bennett, D. (2017, 15 November). The Brutal Fight to Mine Your Data and Sell It to Your Boss. Bloomberg. <https://www.bloomberg.com/news/features/2017-11-15/the-brutal-fight-to-mine-your-data-and-sell-it-to-your-boss>
- Broucke, V. S., & Baesens, B. (2018). Practical Web Scraping for Data Science: Best Practices and Examples with Python. Apress.
- Cavallo, A. (2016, 28 marzo). The Billion Prices Project: Using Online Prices for Measurement and Research. NBER. <https://www.nber.org/papers/w22111>
- CONSEJO NACIONAL DE ACREDITACION. (s.f.). Principios de la acreditación. CNA. Consultado el 20 de noviembre de 2022 https://www.cna.gov.co/1779/w3-article-402545.html?_noredirect=1)%20%20((Tomado
- CRUMMY. (s.f.). Beautiful Soup: We called him Tortoise because he taught us. Consultado el 8 de diciembre de 2022. <https://www.crummy.com/software/BeautifulSoup/>
- CSS. (s.f.). ¿Qué es CSS? Lenguaje CSS. Consultado el 28 de noviembre de 2022. <https://lenguajecss.com/css/introduccion/que-es-css/>
- DJANGO. (s.f.). Why Django. Django overview. Consultado el 29 de noviembre de 2022. <https://www.djangoproject.com/start/overview/>
- DJANGO-REST-FRAMEWORK. (s.f.). Home - Django REST framework. Consultado el 30 de noviembre de 2022. <https://www.django-rest-framework.org/>
- DOCKER. (s.f.). About Docker. Consultado el 1 de diciembre de 2022 <https://www.docker.com/company/>
- Fleitas, F. (2017). La Inteligencia Artificial e Ingeniería del Conocimiento. Alianza Editorial.

GITHUB. (s.f.). About GitHub. Consultado el 10 de enero de 2023 <https://github.com/about>

HTML. (s.f.). ¿Qué es HTML? Lenguaje HTML. Consultado el 18 de noviembre de 2022. <https://lenguajehtml.com/html/introduccion/que-es-html/>

JSON. (s.f.). Introducción a JSON. <https://www.json.org/json-es.html>

Lasa Gómez, Carmen, Álvarez García, Alonso, & Heras del Dedo, Rafael de las. (2017). Métodos Ágiles. Anaya Multimedia.

MINISTERIO DE EDUCACIÓN NACIONAL. (2017, 7 septiembre). SNIES Sistema Nacional de Información de la Educación Superior. SNIES. Consultado el 18 de noviembre de 2022. <https://snies.mineduacion.gov.co/portal/EL-SNIES/Que-es-el-SNIES/>

Monte Galiano, Josep. (2016). Implantar scrum con éxito. Editorial UOC.

ORACLE. (s.f.). ¿Qué es la inteligencia artificial (IA)? Consultado el 12 de diciembre de 2022. <https://www.oracle.com/co/artificial-intelligence/what-is-ai/>

Orallo, J. H., Quintana, M. J. R., & Ramírez, C. F. (2004). Introducción a la minería de datos (1.a ed.). Pearson Education.

PHP. (s.f.). PHP. ¿Qué es PHP? Consultado el 30 de noviembre de 2022 <https://www.php.net/manual/es/intro-what-is.php>

PYPI. (s.f.). python-crontab. Consultado el 8 de enero de 2023 <https://pypi.org/project/python-crontab/>

PYTHON. (2008, Agosto.). About Python. Python. Consultado el 28 de noviembre de 2022. <https://www.python.org/about/>

Qian Liu, Bei Chen, Jiaqi Guo, Morteza Ziyadi, Zeqi Lin, Weizhu Chen, Jian-Guang Lou. (2022). International Conference on Learning Representations. Consultado el 13 de enero de 2023. <https://openreview.net/forum?id=O50443AsCP>

SCRAPY. (s.f.). Scrapy 2.7 documentation. Consultado el 28 de noviembre de 2022. <https://docs.scrapy.org/en/latest/>

SELENIUM. (s.f.). Webdriver. Webdriver Selenium. Consultado el 29 de noviembre de 2022.
<https://www.selenium.dev/documentation/webdriver/>

Sprajcar, S. (2016). Using big data to predict suicide risk among Canadian youth. SAS.
https://www.sas.com/en_ca/insights/articles/analytics/using-big-data-to-predict-suicide-risk-canada.html