Caracterización de nuevos estudiantes en la Universidad de Boyacá, entre 2020 y 2023, mediante analítica de datos con Python

Maicol Daniel Rojas Martinez

Universidad de Boyacá

Facultad de Ciencias e Ingeniería

Programa de Ingeniería de Sistemas

Tunja

2024

Caracterización de nuevos estudiantes en la Universidad de Boyacá, entre 2020 y 2023, mediante analítica de datos con Python

Maicol Daniel Rojas Martinez

Trabajo de grado de semillero de investigación SISDYTEL para optar al título de:
Ingeniero de Sistemas

Directora:
Clara Patricia Avella Ibáñez
Ingeniera de Sistemas

Universidad de Boyacá
Facultad de Ciencias e Ingeniería
Programa de Ingeniería de Sistemas
Tunja
2024

Nota de aceptación

_____

_____

_____

_____

_____

Firma presidente del jurado

_____

Firma del jurado

_____

Firma del jurado

Tunja, 21 de mayo de 2024

“Únicamente el graduando es responsable de las ideas expuestas en el presente trabajo”. (Lineamientos constitucionales, legales e institucionales que rigen la propiedad intelectual).

Contenido

Lista de Figuras

Pág.

Lista de Tablas

Pág.

Lista de Anexos

Pág.

Glosario

Analítica de datos: proceso de convertir datos sin procesar en información práctica, que incluye una serie de herramientas, tecnologías y procesos para encontrar tendencias y resolver problemas mediante datos (Amazon Web Services, s.f.).

Análisis exploratorio de datos: método para analizar y sintetizar conjuntos de datos y resumir sus características principales, comúnmente empleado en métodos de visualización de datos (IBM, s.f).

Api: mecanismo que permite a dos componentes de software comunicarse entre sí mediante un conjunto de definiciones y protocolos La arquitectura de las API suele explicarse en términos de cliente y servidor. La aplicación que envía la solicitud se llama cliente, y la que envía la respuesta se llama servidor (Amazon Web Services, s.f.).

Big data*:* conjunto de estrategias, tecnologías y sistemas para el almacenamiento, procesamiento, análisis y visualización de conjuntos de datos complejos (Casas Roma et al., 2019).

Data frame*:* objeto habitual para el almacenamiento de conjuntos de datos. En este tipo de objetos cada individuo de la muestra se corresponde con una fila y cada una de las variables con una columna. Similar a una matriz con la ventaja que permiten que los valores de las distintas columnas sean de tipos diferentes (Fernandez et al., 2023).

Http*:* protocolo de la capa de aplicación para la transmisión de documentos hipermedia, como HTML. Diseñado para la comunicación entre los navegadores y servidores web, aunque se puede utilizar para otros propósitos también. Sigue el clásico modelo cliente-servidor, en el que un cliente establece una conexión con el servidor, realiza una petición y espera hasta que recibe una respuesta de este (MDN Web Docs, 2023).

***Hosting**:* servicio online que hace que se pueda acceder a un sitio web en internet. Es decir, es un espacio en un servidor que almacena todos los archivos y datos de un sitio para que funcione correctamente (Bustos, 2024).

JavaScript: lenguaje de programación o de secuencias de comandos que permite implementar funciones complejas en páginas web, cada vez que una página web hace algo más que sentarse allí y mostrar información estática (MDN Web Docs, s.f.).

***Json**:* formato ligero de intercambio de datos. json es de fácil lectura y escritura para los usuarios y fácil de analizar y generar por parte de las máquinas. JSON se basa en un subconjunto del lenguaje de programación JavaScript (JSON, s.f.).

***Python**:* lenguaje de programación interpretado, de alto nivel y enfocado principalmente a la legibilidad y facilidad de aprendizaje y uso (Hinajosa Gutierrez, 2015).

***Railway**:* plataforma de implementación diseñada para agilizar el ciclo de vida del desarrollo de software, comenzando con implementaciones instantáneas y escalabilidad sin esfuerzo, extendiéndose a integraciones CI/CD y observabilidad integrada (Railway, s.f.).

***React**:* biblioteca de JavaScript para renderizar interfaces de usuario (UI). La UI se construye a partir de pequeñas unidades como botones, texto e imágenes, combinándolas en componentes reutilizables y anidables (React, s.f.).

Url; dirección que es dada a un recurso único en la Web. En teoría, cada URL válida apunta a un único recurso. Dichos recursos pueden ser páginas HTML, documentos CSS, imágenes, etc (MDN Docs, s.f.).

***Visualización de datos**:* representación gráfica de los datos mediante el uso de gráficos comunes, como cuadros, diagramas, infografías y animaciones, lo cual facilita la comprensión de estos (IBM, s.f.).

Resumen

Caracterización de nuevos estudiantes en la Universidad de Boyacá, entre 2020 y 2023 mediante analítica de datos con Python:

El presente documento integra los resultados obtenidos durante el desarrollo de este proyecto de análisis de datos que englobo las fases de: comprensión de los datos, preparación de los datos, modelado, desarrollo del sitio web, evaluación y despliegue. El propósito de este análisis es examinar la información relacionada con la caracterización de nuevos estudiantes que ingresan a la Universidad de Boyacá, suministrada por DIPA.

El objetivo general del proyecto es Desarrollar una aplicación web en Python que presente, mediante un modelo de visualización de datos, la caracterización de los estudiantes nuevos que ingresaron a la Universidad de Boyacá, entre los años 2020 y 2023.

El desarrollo del proyecto se llevó a cabo con la implementación de la metodología CRISP-DM, ampliamente utilizada en el desarrollo de proyectos de análisis de datos y que posee las siguientes fases: compresión del negocio, interpretación de los datos, preparación de los datos, modelado, evaluación y despliegue.

Como conclusión general, se corrobora con la experiencia adquirida durante el desarrollo de este proyecto la viabilidad de implementar un sistema de análisis de datos en la Universidad de Boyacá utilizando las herramientas de Python y la biblioteca Pandas.

Palabras claves: Analítica de datos, caracterización, Python, limpieza de datos, Pandas, visualización de datos.

Abstract

Characterization of new students at the University of Boyacá, between 2020 and 2023 using data analytics with Python:

This document integrates the results obtained during the development of this data analysis project, which encompasses the phases of data understanding, data preparation, modeling, website development, evaluation, and deployment. The purpose of this analysis is to examine the information related to the characterization of new students entering the University of Boyacá, provided by DIPA.

The general objective of the project is to conduct an exploratory and descriptive analysis of the characterization data of new students entering the University of Boyacá, between 2020 and 2023.

The development of the project was carried out with the implementation of the CRISP-DM methodology, which is widely used in the development of data analysis projects and has the following phases: business understanding, data interpretation, data preparation, modeling, evaluation and deployment.

As a general conclusion, the viability of implementing a data analysis system at the University of Boyacá using Python tools and the Pandas library is confirmed based on the experience gained during the development of this project.

Keywords: Data analytics, characterization, Python, data cleaning, Pandas, data visualization.

Introducción

El presente proyecto contiene el resultado de la aplicación de técnicas y procesos de análisis de datos a la información relacionada con la caracterización de nuevos estudiantes de la Universidad de Boyacá entre los años 2020 y 2023. Se tuvo como objetivo la generación de un modelo que permitiera categorizar y mostrar de manera visual los datos recolectados, con el propósito que los directivos de la Universidad, afines a estos tengan la capacidad de identificar con base en el modelo implementado, patrones de comportamiento en los datos a nivel semestral y anual, como a nivel histórico respectivamente, procurando que la identificación de estos patrones apoye la toma de decisiones en procesos de mercadeo y publicidad, así como el mejoramiento y adaptación en programas de alerta temprana que realiza el bienestar universitario.

La iniciativa para realizar el proyecto nació en la División de Planeación y Acreditación (DIPA), desde donde se formularon los alcances del proyecto y se suministró el conjunto de datos que fue almacenado, procesado y analizado; además, participó en la validación del modelo visual implementado. De esta manera el proyecto sienta sus bases a partir de la identificación de las necesidades de la dependencia relacionada, mencionada anteriormente, correspondiente al análisis de datos históricos de los datos recolectados y almacenados semestralmente en relación con los estudiantes nuevos que ingresan a la Universidad de Boyacá. Posterior a la identificación de los requerimientos se procedió con la obtención, compresión, limpieza y depuración de los datos para permitir la creación y aplicación de modelos de análisis sobre la información requerida, con la finalidad de generar un modelo visual gráfico que facilite la interpretación de estos datos.

Para ello, se utilizó la metodología CRIPS-DM (Proceso estándar cruzado para minería de datos), la cual es ampliamente reconocida por la comunidad de analistas y científicos de datos para llevar a cabo proyectos de análisis de datos, debido a su enfoque, estructura, ya que proporciona una hoja de ruta clara para el éxito de este tipo de proyectos. (Arias, 2023)

Para el desarrollo de este proyecto se empleó como apoyo el trabajo de tesis titulado "**Modelo de Procesos para Elicitación de Requisitos en Proyectos de Explotación de Información**" (Pollo Cattaneo, 2018), en el cual se explica claramente cómo se realiza un proyecto de análisis de datos y se definen algunos formatos para complementar el modelo CRISP-DM. De estos formatos, en particular se utilizó la matriz de interesados del proyecto, la cual permite identificar los actores implicados en el desarrollo y resultados del proyecto.

Este documento se ha estructurado en capítulos, donde cada uno representa los resultados de cada objetivo propuesto en el anteproyecto. Los capítulos incluyen: resultados esperados a partir del análisis exploratorio, donde se realizó un análisis del funcionamiento actual del negocio, así como la identificación del problema y los requerimientos del proyecto; exploración y preparación de los datos, que detalla el proceso de obtención, análisis y limpieza de los datos para su estandarización; diseño del modelo de datos, que expone la estructura del modelo desarrollado; desarrollo de la aplicación del software, que describe el proceso de implementación del software, así como las funcionalidades que este presenta; y validación y despliegue del modelo, donde se expone la retroalimentación dada por parte de la División de Planeación y Acreditación (DIPA) y el proceso realizado para subir el sitio web a un hosting. Adicionalmente, se exponen las conclusiones, recomendaciones y referencias bibliográficas. El anteproyecto se adjunta en el anexo A.

Resultados esperados a partir del análisis exploratorio y descriptivo de los datos de caracterización

En este capítulo, se aborda la fase de compresión del negocio, según la metodología CRISP-DM. Durante esta se realizó un análisis exhaustivo del funcionamiento actual del negocio, así como la identificación del problema actual el cual pretende abordar este proyecto. Además, se han llevado a cabo una serie de actividades adicionales que se detallarán a continuación.

Comprensión del negocio

La finalidad de esta fase inicial era realizar la interpretación de los objetivos y requerimientos que sustentaban al proyecto desde la visión de la organización, en este caso, la Universidad de Boyacá a través de la dependencia de DIPA; se identificaron los requerimientos y se fijaron los objetivos de la exploración de los datos.

Definición del objetivo del proyecto

Durante el desarrollo de este proceso se fijó el objetivo de establecer cuáles eran las divisiones involucradas en el desarrollo del proyecto, su propósito y como este se alineaba con el Plan Estratégico de la Universidad de Boyacá 2022-2029, específicamente con el objetivo estratégico de “Implementar la Universidad 4.0” a través del proyecto denominado “Analítica de datos para los procesos académicos”. De igual manera este proyecto está alineado con el Plan de Desarrollo Institucional de la Universidad de Boyacá 2022-2025, enmarcado en la “Política de Desarrollo Tecnológico”, la cual tiene como objetivo fortalecer el desarrollo y promover la cultura en tecnologías de la información y comunicación, buscando apoyar los procesos de docencia, investigación y proyección social. Adicionalmente, con este proyecto se busca respaldar la gestión académica y administrativa de la universidad con el fin de brindar procesos eficientes que busquen aumentar la calidad de la institución a nivel general.

Los interesados del proyecto fueron perfilados con ayuda de la directora del proyecto y en función al rol que desempañaban en las dependencias implicadas se logró identificar el problema de investigación. Como soporte se diligencia el Formato de la tabla 1.

Tabla 1

Formato interesados del proyecto

Interesados del proyecto					
Analista	Maicol Daniel Rojas Martinez		Fecha		22/04/2023
Posición	**Organización Sector**	**Rol en el Proyecto**	**Datos Contacto**		
			Nombre	**Email**	**Teléfono**
Directora DIPA desde febrero de 2020 hasta diciembre de 2022	Universidad de Boyacá (División de planeación y acreditación)	Interesado	Clara Patricia Avella Ibáñez	cpavella@uniboyaca.edu.co	+57 608 7450000 Ext. 15513
Directora DIPA hasta desde enero de 2023 hasta la fecha	Universidad de Boyacá (División de planeación y acreditación)	Interesado	Sonia Milena Forero Ropero	dipa@uniboyaca.edu.co	+57 608 7450000 Ext. 15513
Directores de programa y tutores	Universidad de Boyacá (Facultades)	Beneficiarios indirectos			
Coordinadora institucional de tutorías	Universidad de Boyacá (División de Bienestar Universitario)	Beneficiaria indirecta	María Carolina Russy Colmenares	mcrussy@uniboyaca.edu.co	+57 608 7450000 Ext. 15104

Fuente: Pollo Cattaneo, M. F. (2018). Modelo de proceso para la elicitación de requerimientos en proyectos de explotación de información. (Tesis de maestría, Universidad Nacional de La Plata). SEDICI - Repositorio institucional de la UNLP. http://sedici.unlp.edu.ar/handle/10915/66760

Teniendo en cuenta que el proyecto se enfoca en una necesidad expresada por parte de la directora de DIPA, hasta diciembre de 2023, Ing. Patricia Avella, se realizaron varias sesiones de asesoría del proyecto con ella, quien explicaba los requerimientos del proyecto y alcances esperados, además de aclarar las dudas que surgían sobre la información de caracterización de estudiantes entre 2020 y 2023.

Investigación del problema y alcance

Al analizar los resultados de las sesiones de asesoría realizadas a la Ing. Patricia Avella, exdirectora de DIPA, se logró la extracción de información relevante para tener un alcance claro y limitado del proyecto. De esta manera se concluyó en que la División de Planeación y Acreditación (DIPA) maneja los procesos relacionados con el almacenamiento y procesamiento de los datos de los estudiantes nuevos que ingresan a la Universidad de Boyacá en cada periodo académico, los cuales son obtenidos a través de una encuesta de caracterización que ellos aplican directamente a los nuevos estudiantes y se complementan con información suministrada por el SIIUB (Sistema Integrado de Información de la Universidad de Boyacá), cuando los aspirantes se inscriben y matriculan. Los datos recolectados y base de estudio en el presente proyecto se encontraban en Excel. DIPA también suministró los informes que genera DIPA por periodo académico, los cuales hasta 2022 se realizaban con generación de gráficas en Excel y análisis que se consolidaba en un documento de texto. En el año 2020 DIPA actualizó la encuesta de caracterización, razón por la cual, el presente proyecto incluye los datos recolectados desde ese año y hasta 2023. En el año 2023 DIPA empezó a generar sus informes de caracterización semestrales con el apoyo de gráficas generadas en dashboards de Microsoft Power Bi, las cuales fueron suministradas para este proyecto, con el fin de conocer la información que se desea procesar, pero de manera histórica desde 2020, ya que desde entonces la información está estandarizada, por el cambio en el instrumento aplicado. El problema actual surge debido a que los informes de caracterización se realizan semestralmente y no se cuenta con un histórico que permita visualizar claramente el comportamiento de la caracterización de los estudiantes, entre un periodo académico y otro, o no se puede observar gráficamente su evolución lo largo del tiempo.

Como complemento a los resultados de las sesiones mencionadas y con el fin de entender lo que se esperaba del proyecto, fueron suministrados los informes de caracterización generados por DIPA en periodos anteriores, los cuales sirvieron de base para conocer las expectativas del proyecto, comprender los datos manejados en los informes y la presentación de estos. Estos informes se pueden observar en el anexo B.

De esta manera el objetivo general y principal criterio para tener en cuenta en este proyecto es la realización de un análisis exploratorio de los datos de caracterización de los nuevos estudiantes que ingresaron a la Universidad de Boyacá, entre los años 2020 y 2023, esto mediante

el diseño e implementación de un modelo de visualización de datos a través de una aplicación de software desarrollada con las tecnologías de Python y React.

El modelo generado para obtener dicho alcance permitió la visualización de los datos por diferentes periodos de tiempo como semestral, anual e histórico, buscando la facilidad para que una persona que lo analice pueda evidenciar fácilmente cómo es el comportamiento de estos en cada periodo de tiempo gráficamente.

Igualmente, se definió que el modelo debería estar en la capacidad de filtrar los datos no solo por periodos de tiempo, sino también por categorías generales como “Información General” e” Información Socioeconómica”; así como categorías específicas que se ajustaron de acuerdo con las variables de estudio más importantes identificadas a través de las sesiones de asesoría y la recapitulación de los informes suministrados como Sede, Programa Académico, etc. Esto con el fin de facilitar la visualización y compresión de los datos al permitir la selección de subconjuntos personalizados en el análisis. Las categorías generales y específicas utilizadas como filtro en el análisis se presentan en las tablas 2 y 3.

Tabla 2

Categorías generales para el filtrado de los datos

Categorías generales
Información General
Información Familiar
Estado Socioeconómico
Educación Previa
Datos Étnicos
Percepción y Satisfacción

Fuente: elaboración propia

Tabla 3

Categorías específicas para el filtrado de los datos

Categorías especificas
Sede
Estrato Socioeconómico
Programa Académico
Sexo
Grupo Étnico

Fuente: elaboración propia

Por otro lado, se descartaron aquellos datos, columnas o variables de estudio que no aportaban ningún valor agregado, eran redundantes o que por limitaciones del alcance del proyecto no se tuvieron en cuenta a la hora de hacer el respectivo análisis. En Tabla 4 se visualizan las variables descartadas para el análisis correspondiente.

Tabla 4

Variables descartadas en el análisis

Variables descartadas
Marca temporal
Nombre completo
Código
Correo electrónico
Sugerencias
Justificación sobre recomendación de la Universidad
Justificación sobre percepción de la Universidad
Factores de tiempo transcurrido desde la incorporación a la universidad
Nombre del medio de comunicación por el cual se enteró de la Universidad de Boyacá

Fuente: elaboración propia

Recolección, exploración y preparación de los datos

En este capítulo, se abordan las fases de compresión y preparación de los datos, según la metodología CRISP-DM. Durante su desarrollo se recopiló y analizó exhaustivamente los datos proporcionados por DIPA, para posteriormente realizar la limpieza y depuración de los datos. A continuación, se presentan los detalles de cada etapa realizada.

Comprensión de los datos

Durante esta etapa, se tenía como objetivo recopilar y analizar la información inicial de los datos que serían fundamentales a fin de establecer las bases para los posteriores procesos de análisis y procesamiento del conjunto de datos suministrado por parte de DIPA.

Recopilación de datos

Como se mencionó anteriormente, para el desarrollo del proyecto, DIPA proporcionó el conjunto de datos a utilizar, estos provenían de tablas en un formato Excel (.xlsx) que contenían la información recolectada proveniente de las respuestas proporcionadas por los estudiantes en la Encuesta de Caracterización realizada semestralmente por parte de la Universidad de Boyacá. El conjunto de datos comprende información desde el segundo semestre del año 2020 hasta el primer semestre del año 2023 y se encuentra divido en archivos semestrales que a su vez se dividen por nivel académico, es decir, pregrado y posgrado. Dichas tablas se utilizaron como la única fuente principal de los datos y, por lo tanto, marcaron el punto de inicio para comenzar con el proceso de analítica de datos. Las tablas suministradas por DIPA se pueden consultar en el Anexo C.

Exploración inicial y calidad de los datos

Durante esta etapa se realizó el acercamiento o exploración inicial con el conjunto de datos proporcionados por DIPA, esto con la finalidad de conocer la estructura, su información almacenada y la calidad de esta, con el objetivo de tener una aproximación real a posibles problemas en la naturaleza y estandarización de los datos.

Estructura y características

Como resultado de realizar la exploración inicial de los datos se encontraron diversos rasgos importantes, que se consideraron y analizaron en busca soluciones para desarrollar las posteriores etapas del análisis de datos sin mayores contratiempos.

El primer rasgo importante es que el conjunto de datos estaba dividido en subconjuntos, es decir, los datos se presentaban separados en tablas alojadas en archivos de Excel individuales categorizados por el semestre, en que los datos fueron registrados a través de las respuestas de la encuesta y el nivel académico de los estudiantes que respondieron esta misma, dicho nivel, cabe recalcar, está definido por pregrado y posgrado, por lo que en conclusión se manejaron alrededor 12 tablas diferentes alojadas individualmente en un archivo de Excel distinto. Con base en esto se consideró que debido a la naturaleza del análisis a realizar no era posible la unión de todos los subconjuntos de datos en un solo, por lo que se consideró trabajar bajo la estructura de categorización actual.

De igual manera, el segundo rasgo encontrado en la estructura de los datos fue que las tablas que almacenaban la información de un semestre y un nivel académico especifico no poseían en conjunto las mismas categorías de información o las variables evaluar en el análisis, esto debido a que DIPA usualmente a nivel semestral o anual actualiza dicha información, tanto con el registro de nuevas categorías, o con la eliminación de antiguas cuando ya no les son de utilidad. Del mismo modo, cabe destacar que la categorización de los datos es diferente en cada nivel académico y que igualmente, hay algunas categorías de datos que se mantienen estándar a lo largo del periodo a evaluar, como en los diferentes niveles académicos. En consecuencia, esto en un principio dificultó la compresión de la estructura de los datos ya que no había una estandarización general, la categorización de los datos era bastante fluctuante y que a su vez estas alojaban un gran número de registros.

De esta manera, para brindar una aproximación real a la exploración realizada a los datos, se revisó en detalle cada tabla, a modo de ejemplo se menciona la tabla que aloja la información de la encuesta de caracterización a estudiantes de pregrado en el segundo semestre del año 2020 alojada en el archivo “DATOS ENCUESTA PREGRADO PRIMER CURSO GENERAL 2020-2” en la cual se tenían registradas 51 preguntas representadas cada una en una columna y que representan las variables a evaluar en el análisis de datos y donde se registraron 166 respuestas.

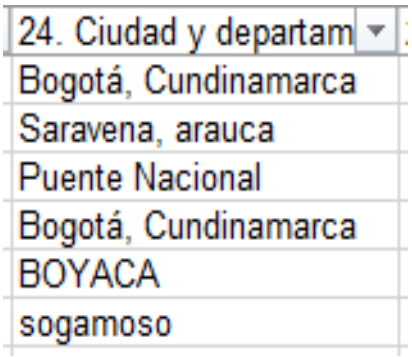
Del mismo, se destaca que durante las sesiones donde se dio avance a este proceso se contó con la participación la directora del proyecto (exdirectora de DIPA), lo que permitió la resolución de las dudas que surgieron a lo largo del proceso acerca de la información y estructura de las diferentes tablas.

Calidad de los datos

Contando con la información general, el propósito de cada tabla y su categoría, se inició la etapa de identificación de la calidad de los datos, con el fin de determinar problemas en la naturaleza y estado de estos. De esta manera, en primer lugar, se estableció que todos los subconjuntos de datos necesitaban una normalización de su formato a minúsculas, así mismo se requería eliminar las tildes, esto con el propósito de evitar la duplicidad o incoherencia de los datos. De otra parte, se encontraron campos vacíos, los cuales se generaron porque no aplicaba la pregunta al estudiante encuestado o porque los estudiantes no respondieron ese campo. También se encontraron inconsistencias ortográficas y de sintaxis en las categorías que no tenían valores predefinidos, sino que estos eran dados por los estudiantes de forma manual como podría ser “Lugar de nacimiento” o “Ciudad de residencia”, así como inconsistencias en el formato de las respuestas prestablecidas en la encuesta donde estas discrepaban en diferentes periodos de tiempo, como por ejemplo las categorías de respuesta de “Estrato Socioeconómico”. Adicionalmente, se hallaron problemas de duplicidad en los campos de “Doble Programa” donde comúnmente si la respuesta era marcada como que el estudiante cursaba doble programa, se seleccionaba como segundo programa el mismo programa al cual pertenecía el estudiante, es decir, no cursaba doble programa. Otra inconsistencia encontrada corresponde a columnas que presentaban duplicidad en algunos periodos de tiempo. Las Figuras 1 a 7 representan ejemplos de algunos escenarios mencionados, encontrados en el conjunto de datos suministrados por DIPA.

Figura 1

Ejemplo de campos sin capitalización del formato de texto

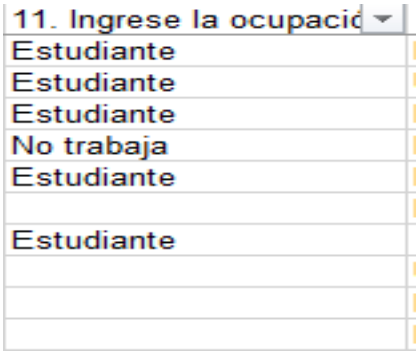


24. Ciudad y departam
Bogotá, Cundinamarca
Saravena, arauca
Puente Nacional
Bogotá, Cundinamarca
BOYACA
sogamoso

Fuente: elaboración propia a partir de información suministrada por DIPA

Figura 2

Ejemplo de campos vacíos o nulos

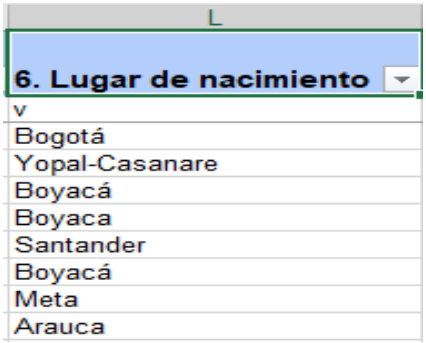


11. Ingrese la ocupaci
Estudiante
Estudiante
Estudiante
No trabaja
Estudiante
Estudiante

Fuente: elaboración propia a partir de información suministrada por DIPA

Figura 3

Ejemplo de errores gramaticales o de ortografía

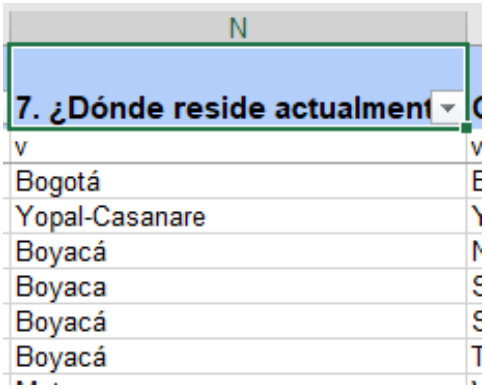


L
6. Lugar de nacimiento
v
Bogotá
Yopal-Casanare
Boyacá
Boyaca
Santander
Boyacá
Meta
Arauca

Fuente: elaboración propia a partir de información suministrada por DIPA

Figura 4

Ejemplo de acentuación o caracteres especiales que generan duplicidad



N
7. ¿Dónde reside actualment
v
Bogotá
Yopal-Casanare
Boyacá
Boyaca
Boyacá
Boyacá

Fuente: elaboración propia a partir de información suministrada por DIPA

Figura 5

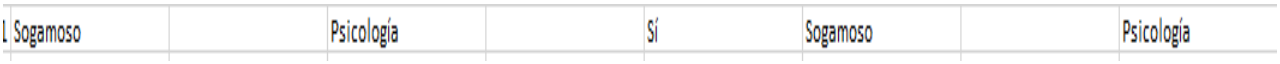
Ejemplo de inconsistencia en el formato de las respuestas preestablecidas para la categoría Estrato socioeconómico 2023 - 2020

EstratoSocioEconomico	EstratoSocioEconomico
3	2 (Bajo)
4	3 (Medio - Bajo)
3	4 (Medio)
No informa	2 (Bajo)
No informa	2 (Bajo)
2	2 (Bajo)
3	2 (Bajo)

Fuente: elaboración propia a partir de información suministrada por DIPA

Figura 6

Ejemplo duplicidad de los datos en las categorías de Programa Académico y Doble Programa

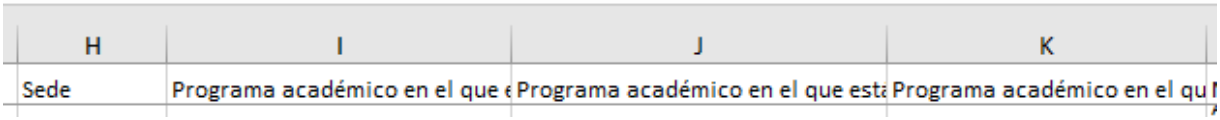


l	Sogamoso		Psicología		Sí	Sogamoso	Psicología

Fuente: elaboración propia a partir de información suministrada por DIPA

Figura 7

Ejemplo duplicidad en las columnas de las tablas



H	I	J	K
Sede	Programa académico en el que e	Programa académico en el que esta	Programa académico en el qu

Fuente: elaboración propia a partir de información suministrada por DIPA

Preparación de los datos

Durante esta etapa se fijó como objetivo la ejecución de todas las tareas relacionadas con la limpieza y estabilización de los datos de cada subconjunto, para lo cual se implementaron técnicas y formatos a los datos recolectados para garantizar la estandarización uniforme de la información con el fin de utilizarlos en las siguientes etapas.

Limpieza y transformación de los datos

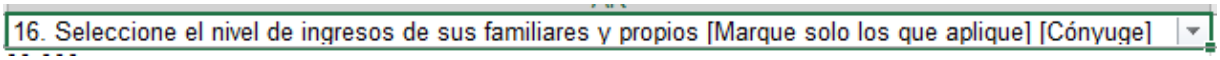
Inicialmente el primer paso en este proceso fue la estandarización y validación de las categorías de cada subconjunto o de cada tabla de datos. Para ello se modificaron los nombres de las columnas, en cada una de las tablas Excel, las cuales inicialmente se encontraban con un formato extendido con la descripción de las diferentes preguntas que tenía la encuesta; por ejemplo: “16. Seleccione el nivel de ingresos de sus familiares y propios [Marque solo los que aplique] [Cónyuge]”, la cual finalmente quedó con un nombre abreviado para su identificación. Con el nombre original (ver Figura 8) de las columnas, se generaban problemas al momento de importar el archivo.xlsx en el script de Python, por lo que se modificó el nombre de cada columna o categoría asignándole uno nuevo en formato camel case y transformándolo en un nombre más corto y adaptable para el script, pero sin perder el valor de referencia de este. Algunos nombres ya estandarizados se pueden observar en la Figura 9.

De la misma manera, una vez estandarizadas las categorías (columnas del archivo Excel) se dio inicio a la normalización del contenido de los datos; para ello se hizo uso de un script en Python y la biblioteca de Pandas, lo cual permitió la homogenización de los datos convirtiendo el texto a minúsculas, eliminando las tildes y los caracteres especiales para evitar duplicidad de los datos como se puede observar en las Figuras 10 y 11. Del mismo modo se realizó la validación

mediante el script de insertar el valor de “No Responde” en los espacios vacíos de la tabla tal como se refleja en las Figuras 12 y 13. Igualmente, se eliminaron algunas categorías o columnas con duplicidad y otras que no iban a formar parte del análisis de datos ya que no estaban estipuladas dentro del alcance de este como se contempla en la Figura 14.

Figura 8

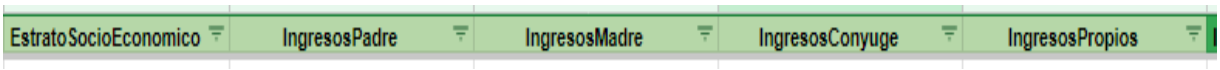
Ejemplo del nombre estándar de las columnas o variables de categorización



Fuente: elaboración propia a partir de información suministrada por DIPA

Figura 9

Ejemplo de los nombres de las columnas estandarizados



Fuente: elaboración propia a partir de información suministrada por DIPA

Figura 10

Script en Python para convertir texto en minúsculas, eliminar tildes y caracteres especiales

```
# Normalizar el texto - Minusculas
for column in df.columns:
    df[column] = df[column].apply(lambda x:x.lower() if isinstance(x, str) else x)

# Normalizar el texto - Omitir acentos / Caracteres especiales
for column in df.columns:
    df[column] = df[column].apply(lambda x: unidecode.unidecode(x) if isinstance(x, str) else x)

# Eliminar espacios innecesarios -> (Xxxxx )
for column in df.columns:
    df[column] = df[column].apply(lambda x: ' '.join(x.split()) if isinstance(x, str) else x)
```

Fuente: elaboración propia

Figura 11

Resultado del Script en Python para convertir texto en minúsculas, eliminar tildes y caracteres especiales


```
"bogota": 10,
"cundinamarca": 8,
"saravena": 1,
"arauca": 1,
"puente nacional": 1,
"boyaca": 21,
"sogamoso": 5,
"risaralda": 1,
"tunja": 12,
```

```
"medicina": 57,
"enfermeria": 20,
"psicologia": 10,
"bacteriologia y lab. clinico": 10,
"comunicacion social": 8,
"terapia respiratoria": 8,
"administracion de negocios internacionales": 8,
"fisioterapia": 6,
"diseno grafico": 5,
"instrumentacion quirurgica": 5,
"arquitectura": 5,
"ingenieria ambiental": 5,
```


Fuente: elaboración propia a partir de la información suministrada por DIPA

Figura 12

Script para validación de espacios vacíos o nulos

```
df = df.fillna('No Responde')
```

Fuente: elaboración propia

Figura 13

Resultado del script de validación de espacios vacíos o nulos


```
"HermanosMenores": {
    "1": 61,
    "0": 60,
    "2": 19,
    "no responde": 13,
    "3": 8,
    "4 o mas": 4
```

```
"SedeDoblePrograma": {
    "no responde": 688,
    "tunja": 1
},
```


Fuente: elaboración propia a partir de la información suministrada por DIPA

Figura 14

Script de exclusión de categorías con duplicidad y no contempladas en el análisis

```
# Columnas excedentes
columns_delete = ['MarcaTemporal','NombreCompleto','Codigo','CorreoElectronico','Sugerencias','JustificacionRecomendacion',
                  'JustificacionPercepcion','NombreMedioComunicacion',
                  'FactoresTiempoTranscurrido']

# Elimina las columnas excedentes si estan presentes
for col in columns_delete:
    if col in df.columns:
        df = df.drop(col,axis=1)
```

Fuente: elaboración propia

Creación del conjunto de datos final

Una vez realizada la etapa de depuración y estandarizado de los datos se realizó la preparación de estos conforme al subconjunto requerido, para ello es importante recalcar que en este proyecto los datos no se exportaron a otro archivo .xlsx una vez finalizada la depuración, sino que los datos ya depurados son fueron guardados en un data fotograma de Pandas, que posteriormente fue importado y utilizado en los diferentes métodos que conforman el modelo de análisis de datos propuesto. En la figura 15 se adjunta un ejemplo de un data frame con los datos estandarizados.

Figura 15

Ejemplo del data frame con los datos depurados almacenados

```
DataFrame
{'ProgramaAcademico': {'medicina': 57, 'enfermeria': 20, 'psicologia': 10, 'bacteriologia y lab. clinico': 10, 'comunic
oterapia': 6, 'diseno grafico': 5, 'instrumentacion quirurgica': 5, 'arquitectura': 5, 'ingenieria ambiental': 5, 'admi
eria sanitaria': 2, 'ingenieria industrial': 2, 'contaduria publica': 1, 'ingenieria mecatronica': 1, 'ingenieria en mu
no': 113, 'masculino': 52}, 'Edad': {18: 40, 17: 36, 19: 25, 20: 17, 21: 11, 16: 7, 24: 7, 22: 6, 23: 6, 27: 4, '30 o m
oHijos': {'no tiene': 153, 1: 8, 2: 4}, 'ConviveCon': {'ambos padres': 70, 'mama': 59, 'solo': 14, 'parientes': 9, 'pap
ersonasCargoEconomicamente': {'no responde': 149, 1: 10, 2: 4, 3: 1, '4 o mas': 1}, 'EstratoSocioEconomico': {'2 (bajo)
'5 (medio - alto)': 1}}
```

Fuente: elaboración propia

Diseño del modelo de datos

En este capítulo, se detalla el proceso de diseño del modelo de datos desarrollado en coherencia con la implementación de la fase de modelado de la metodología CRISP-DM, con el objetivo de dar respuesta a las necesidades específicas de información expresadas por DIPA, de la Universidad de Boyacá.

Modelado

Esta etapa se centró en el diseño e implementación de un modelo de clasificación de información, con el propósito de categorizar y realizar un análisis exploratorio a la información almacenada por periodos de tiempo semestrales, anuales e históricos.

Para ello, el modelo se desarrolló con las tecnologías de Python y su biblioteca Pandas, además, de integrar la biblioteca ChartsJs para la creación de gráficas dinámicas, incluyendo diferentes tipos como pie, lineal y bar, lo que ha permitido una visualización clara y efectiva de los resultados. Esta elección de tecnologías se ha fundamentado en la búsqueda de un modelo de categorización robusto y escalable, así como la creación del modelo de visualización intuitivo y accesible que facilite la comprensión y exploración de los datos de manera eficiente.

Selección de técnicas de modelado

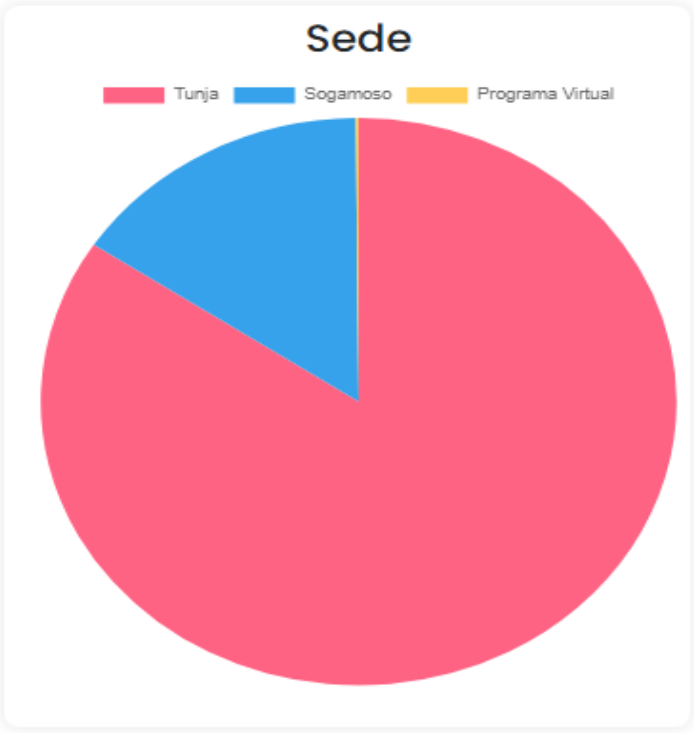
En el desarrollo de este proyecto se implementó un modelo de clasificación visual con el objetivo de categorizar la información correspondiente a la caracterización de los estudiantes por periodos semestrales y anuales como por el histórico total de los datos. Este enfoque hace uso principalmente de técnicas de visualización de datos, buscando promover la precisión y calidad de la presentación gráfica. De este modo la visualización de los datos se presentó a través de gráficas importadas y personalizadas desde la librería ChartsJs tales como gráficas de líneas, gráficas de barras y gráficas de tortas.

Generación de gráficos con Python, Pandas y ChartsJs

En esta sección del proyecto, se presenta el proceso de diseño e implementación del modelo de análisis de datos con el objetivo de clasificar y explorar la información almacenada. Para ello, se han emplearon las herramientas de Python y Pandas, las cuales permitieron realizar un análisis exhaustivo de los datos, abordando procesos específicos como la limpieza y depuración de los datos, seguido de la creación de un nuevo conjunto de datos con estos ya depurados, así culminando con la aplicación de diversos métodos estadísticos para realizar conteos, calcular promedios y comparar los datos. La elección de Python y Pandas se basó en su versatilidad y eficacia en el manejo de datos, lo que permitió un desarrollo ágil y escalable del modelo visual. Además, de integrar la librería ChartsJs para la generación de gráficas dinámicas, que ofrecen una representación visual clara y comprensible de los resultados del análisis, como se puede observar en las Figuras 16 y 17. Estas gráficas reflejan de manera satisfactoria el potencial del análisis, cumpliendo con los lineamientos establecidos. De esta manera, los gráficos son fáciles de interpretar y ofrecen la alternativa de ocultar o mostrar categorías de datos, lo que mejora la legibilidad y la comprensión de la información presentada en escenarios donde haya un gran cumulo de datos como en las variables de estudio de “Programa Académico” o “Convive con”. Esta funcionalidad se puede observar en las Figuras 18 y 19. Además, se ha implementado un elemento "tooltip" que presenta los detalles de los datos de forma clara y concisa, como se muestra en la Figura 20. Esta combinación de elementos visuales y herramientas interactivas garantiza una experiencia de análisis efectiva para los usuarios.

Figura 16

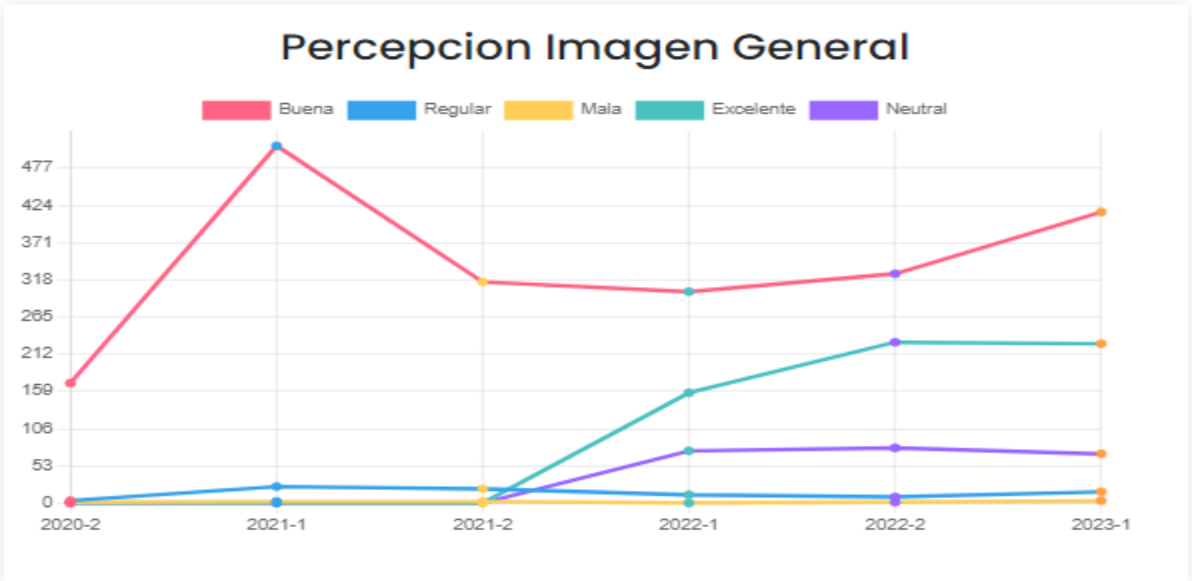
Gráfica generada con la librería ChartsJs de tipo pastel



Fuente: elaboración propia a partir de la información suministrada por DIPA

Figura 17

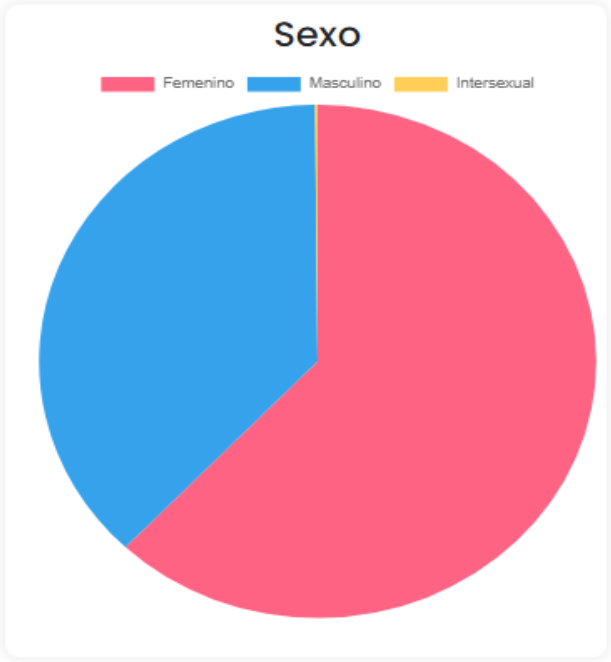
Gráfica generada con la librería ChartsJs de tipo lineal



Fuente: elaboración propia a partir de la información suministrada por DIPA

Figura 18

Ejemplo escenario gráfica con todos sus valores activos



Fuente: elaboración propia a partir de la información suministrada por DIPA

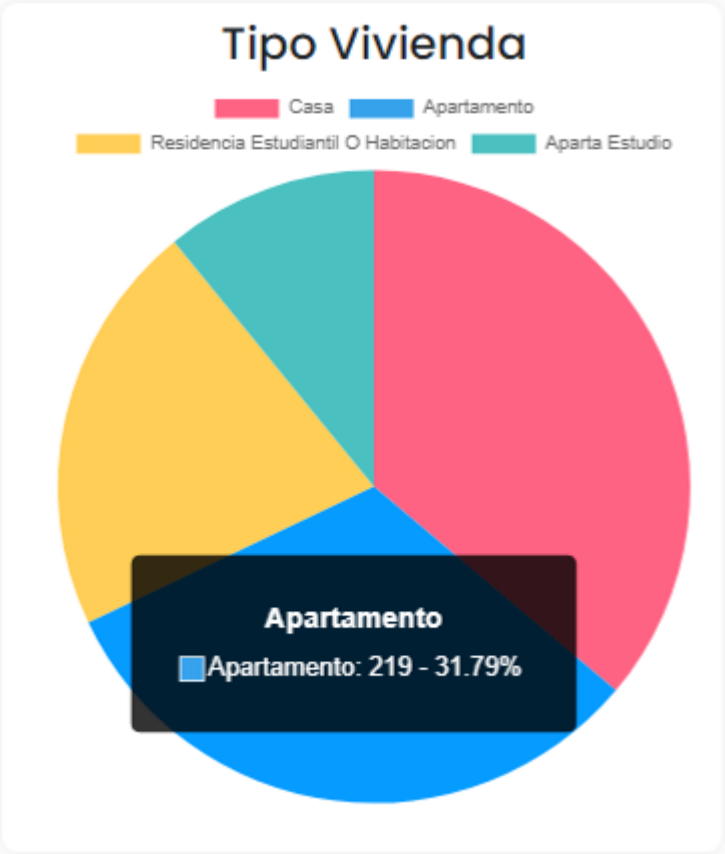
Figura 19

Ejemplo escenario gráfica con algunos valores ocultos

Fuente: elaboración propia a partir de la información suministrada por DIPA

Figura 20

Elemento tooltip que detalla la información de los datos



Fuente: elaboración propia a partir de la información suministrada por DIPA

Creación de las gráficas

El proceso de creación de las gráficas se inició importando los archivos necesarios, que son cargados en un data frame utilizando Python y Pandas. Una vez en el data frame, se llevaron a cabo los procesos de limpieza y depuración de los datos para garantizar su integridad y calidad (mencionado anteriormente). Posteriormente, se validó el data frame resultante para asegurar su coherencia y se exportó a la clase encargada de realizar los métodos estadísticos, como el conteo, el cálculo de promedios y la comparación, entre otros. Los resultados de estos métodos se almacenaron en un diccionario, que se exportó como un objeto JSON. Este objeto JSON es consumido en el cliente, donde se han creado componentes específicos para cada tipo de gráfica. Los datos alojados en el JSON son utilizados por estos componentes para generar dinámicamente

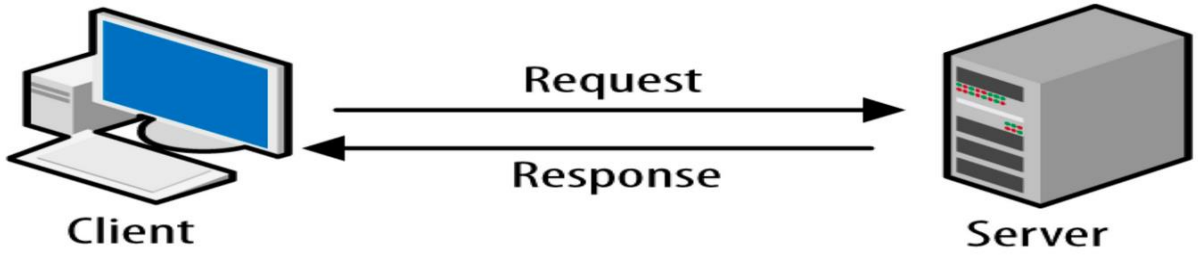
las gráficas necesarias, proporcionando así una representación visual clara y comprensible de los datos depurados. De esta manera, el proceso garantizó la integración fluida entre el análisis realizado en Python y la visualización de los resultados en el cliente con React, utilizando la librería ChartsJs para crear gráficas personalizadas según las necesidades del proyecto.

Desarrollo de la aplicación de software

En este capítulo se describe el desarrollo de la aplicación web que permite visualizar el modelo gráfico creado en la fase anterior; para ello se emplearon diversas tecnologías que permitieron crear una plataforma robusta, escalable y modular para el manejo eficiente de la información. Se utilizó Python como lenguaje principal de programación para el servidor, aprovechando su versatilidad y amplio soporte en el ámbito de la ciencia de datos. Para la creación del cliente, se decidió utilizar React, un framework de JavaScript ampliamente reconocido por su eficiencia y facilidad de uso en la construcción de interfaces de usuario dinámicas. Del mismo modo, se implementó Django Rest Framework como framework de desarrollo web en el lado del servidor, proporcionando una sólida base para la creación de la API de datos y la API de autenticación. Estas APIS permiten la comunicación entre el cliente y el servidor, asegurando un intercambio seguro y eficiente de información. Además, se empleó SQLite y PostgreSQL como base de datos para almacenar los usuarios y sus credenciales de autenticación, garantizando un almacenamiento seguro y eficiente de la información sensible. En cuanto a la arquitectura, se adoptó un enfoque cliente-servidor, donde el cliente desarrollado en React se comunica con el servidor Django a través de solicitudes HTTP. La Figura 21 representa la arquitectura del proyecto y el código de este se adjunta en el Anexo D.

Figura 21

Arquitectura del proyecto



Fuente: Cotzo, J. (2021, 15). *Patrones de arquitectura de software*. *https://juliocotzo.medium.com/patrones-de-arquitectura-de-software-6cffda7dd39e*

Estructura del servidor

Estructura y componentes de la API de datos

El API de datos constituye el pilar fundamental del proyecto, siendo el punto central donde se lleva a cabo todo el análisis de datos. Con el objetivo de estructurar la API de datos de manera eficiente y escalable, se trabajó bajo un enfoque modular. De esta manera, se definieron paquetes individuales para el reporte por período y el reporte histórico, reconociendo que estos no comparten suficientes atributos como para ser reutilizables entre sí. Asimismo, se creó un paquete "core" que almacena la información compartida por los diferentes paquetes de la API. Dentro de este paquete, se incluyeron clases como "Load_Data", "Items", "Paths", "Methods" y "Items_Filtered", que desempeñan funciones clave como cargar y depurar los datos, almacenar los nombres de las columnas o variables para categorizar los datos, manejar los directorios de los archivos Excel, implementar los distintos métodos utilizados en los procesos de análisis de datos y filtrar los datos según los criterios seleccionados. Además, en la API se han definido las URL de los diferentes endpoints correspondientes a cada funcionalidad, así como las vistas que capturan los parámetros enviados desde el cliente para ejecutar el análisis de datos de manera adecuada. Este enfoque estructurado y modular garantiza una gestión eficiente y organizada de los datos, lo cual facilita su procesamiento y análisis.

Funcionamiento del API de datos en la aplicación

El servidor recibe la solicitud HTTP enviada desde el cliente de React y la evalúa para determinar su naturaleza. Esta solicitud es entonces enviada al método pertinente de la vista proporcionada por Django Rest Framework. Esta vista dirigirá automáticamente los parámetros recibidos a la clase principal encargada de manejar el ruteo correspondiente. A partir de estos parámetros, se realiza una validación del nivel académico, así como del periodo de análisis, ya sea semestral, histórico o anual. Posteriormente, basándose en la validación del periodo, se busca el ciclo especificado para su análisis. De esta manera, se procede a cargar los archivos Excel necesarios para los requerimientos específicos, posterior a esto se realiza la depuración y estandarización de los datos contenidos en las tablas, para luego alojar los resultados de la limpieza

en un data frame. Seguidamente, se verifica si se ha proporcionado alguna categoría específica como parámetro, o algún requerimiento que se utilice como filtro de datos. Con esto validado, el data frame se pasa al método correspondiente de análisis, ya sea conteo, conteo histórico, promedio, moda, entre otros. Finalmente, el resultado del análisis se devuelve al cliente en forma de objetos JSON como se puede evidenciar en la Figura 22, proporcionando así la información solicitada de manera clara y estructurada. La arquitectura de la API de datos se presenta en la Figura 23.

Figura 22

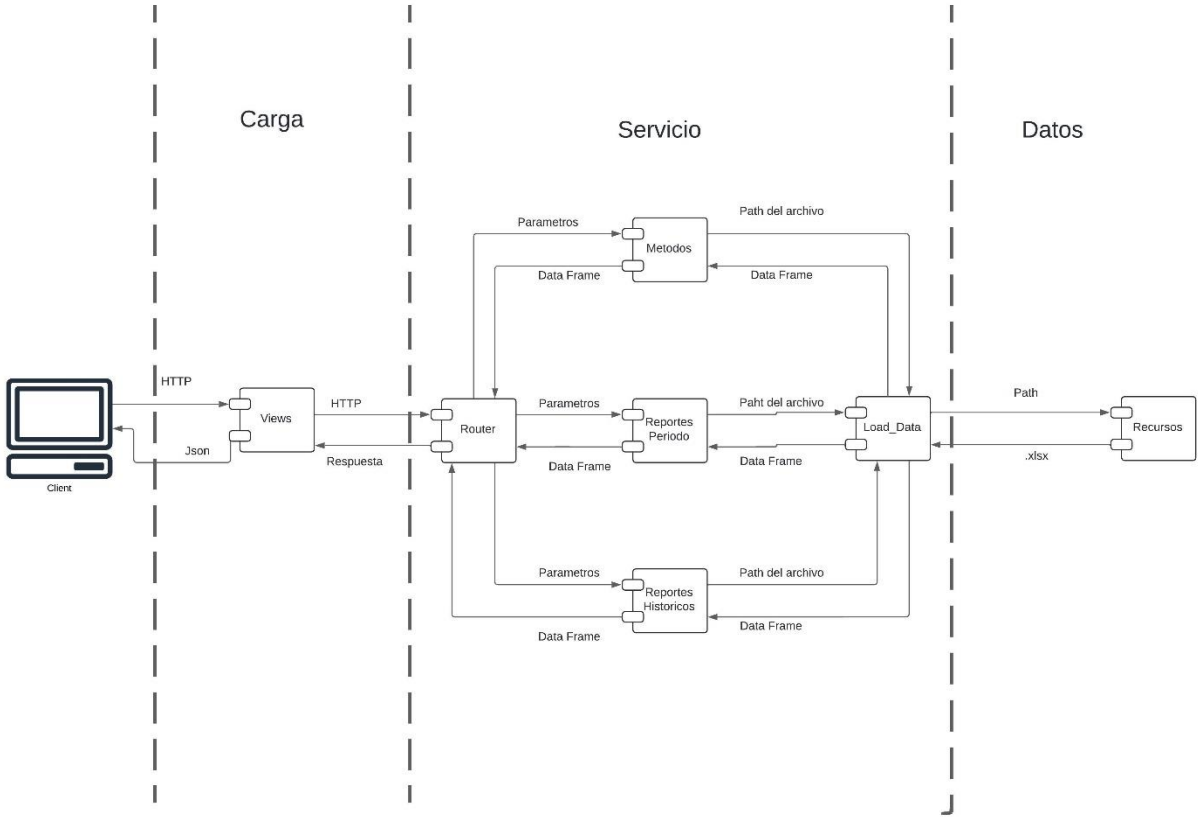
Ejemplo del JSON resultante del análisis de datos


```
{
    "Sede": {
        "tunja": 581,
        "sogamoso": 107,
        "programa virtual": 1
    },
    "ProgramaAcademico": {
        "medicina": 136,
        "enfermeria": 62,
        "psicologia": 52,
        "fisioterapia": 48,
        "diseno grafico": 46,
        "bacteriologia y laboratorio clinico": 43,
        "derecho y ciencias politicas": 43,
        "ingenieria de sistemas": 40,
        "arquitectura": 37,
        "ingenieria mecatronica": 30,
        "administracion de negocios internacionales": 26,
        "ingenieria industrial": 25,
        "administracion de empresas": 22,
        "terapia respiratoria": 18,
        "ingenieria multimedia": 18,
        "comunicacion social": 13,
        "ingenieria ambiental": 10,
        "ingenieria civil": 8,
        "licenciatura en pedagogia infantil": 6,
    "CantidadLectura": {
        "entre 1 y 2 libros": 421,
        "ningun libro": 118,
        "entre 3 y 5 libros": 112,
        "mas de 5 libros": 38
    },
    "FrecuenciaFumar": {
        "nunca": 578,
        "esporadicamente": 70,
        "todos los dias": 18,
        "una vez por semana": 12,
        "cada mes": 7,
        "cada quince dias": 4
    },
    "FrecuenciaBebidasAlcoholicas": {
        "esporadicamente": 305,
        "nunca": 270,
        "cada mes": 56,
        "una vez por semana": 31,
        "cada quince dias": 25,
        "todos los dias": 2
    },
    "FrecuenciaSustanciasPsicoactivas": {
        "nunca": 668,
        "esporadicamente": 15,
        "una vez por semana": 3,
        "cada mes": 3
    },
```


Fuente: elaboración propia a partir de la información suministrada por DIPA

Figura 23

Arquitectura de la API datos



Fuente: elaboración propia

Estructura del cliente

Posterior a la implementación del servidor, el cliente se diseñó e implementó haciendo uso de la biblioteca de React, debido a que es una biblioteca dedicada exclusivamente al desarrollo de interfaces de usuario de manera simple y estandarizada, que brinda la capacidad de crear interfaces escalables y reutilizables a través de componente. Además, esta biblioteca brinda una mayor compatibilidad con la librería ChartsJs, la cual se utilizó para la creación de los gráficos del modelo visual del análisis.

La estructura manejada en el cliente se basa en la implementación de módulos y componentes. Cada módulo representa una sección del sitio web, que a su vez contiene páginas como la de reportes por periodo, reporte de inicio e inicio. Teniendo a su vez en cada módulo

componentes propios y compartidos; los componentes compartidos son aquellos utilizados en diferentes módulos de la aplicación.

Los módulos se describen a continuación:

Reportes por periodo

El módulo de reportes por período está diseñado para facilitar el análisis de datos sobre los nuevos estudiantes en la Universidad. Contiene un menú lateral que permite la navegación entre los diferentes modos de reporte, incluyendo General, Semestral y Anual, así como un reporte histórico para una visión a largo plazo. Además, cuenta con una barra de herramientas dinámica que brinda al usuario la capacidad de filtrar los datos según sus necesidades específicas. Esta barra de herramientas permite seleccionar el nivel académico, el período de análisis, el ciclo de estudio, la categoría de los datos y el punto de referencia para filtrar los datos de manera precisa. Con esta funcionalidad, los usuarios pueden personalizar sus análisis para obtener información relevante y detallada sobre la caracterización de los nuevos estudiantes, lo que facilita la toma de decisiones informadas y la identificación de tendencias significativas en diferentes períodos de tiempo. En las figuras 24, 25 y 26 se puede visualizar el menú de navegación, los elementos desplegables para filtrar la información y la barra de herramientas respectivamente.

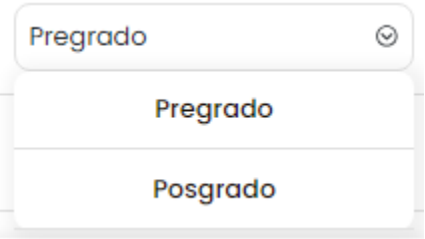
Figura 24

Menú lateral de navegación

Fuente: elaboración propia

Figura 25

Elemento dropdown para filtrar el análisis de datos por nivel académico



Fuente: elaboración propia

Figura 26

Elemento dropdown para filtrar el análisis de datos por categorías generales

Fuente: elaboración propia

Figura 27

Elemento dropdown para filtrar el análisis de datos por una categoría específica

Fuente: elaboración propia

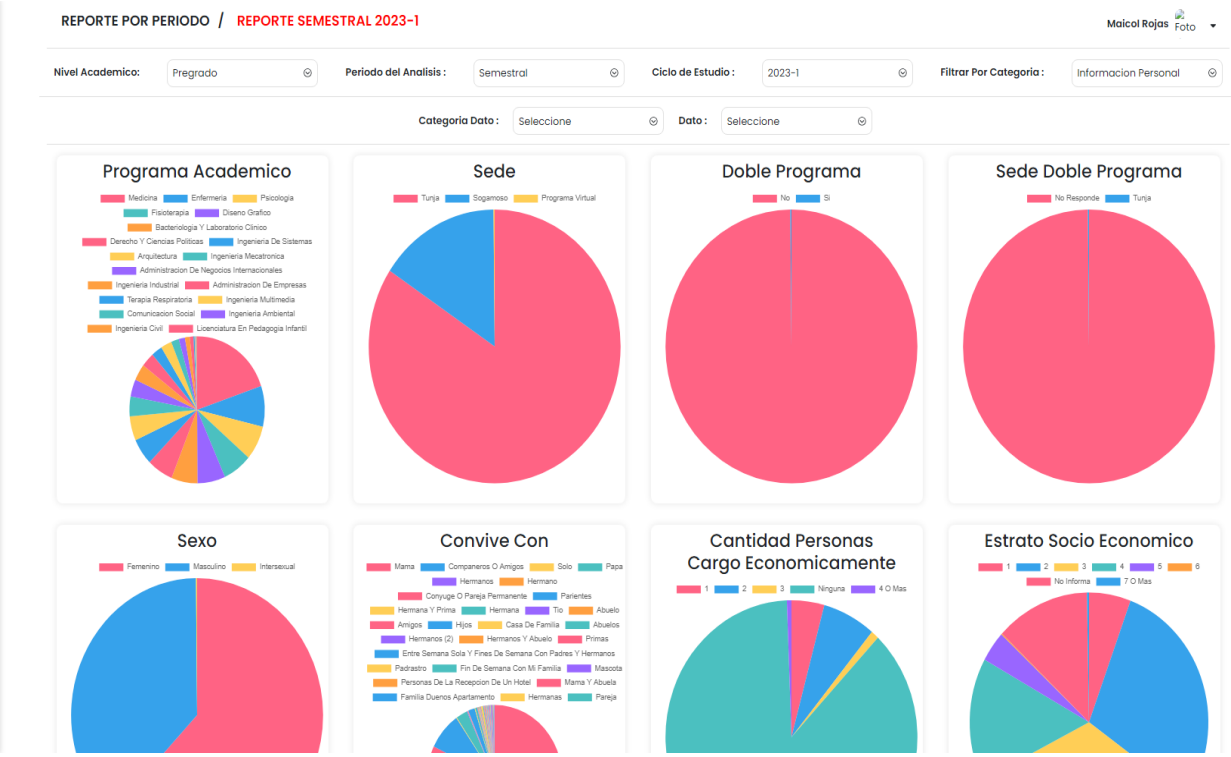
Figura 28

Barra de herramientas

Fuente: elaboración propia

Figura 29

Vista general del módulo reportes por periodo



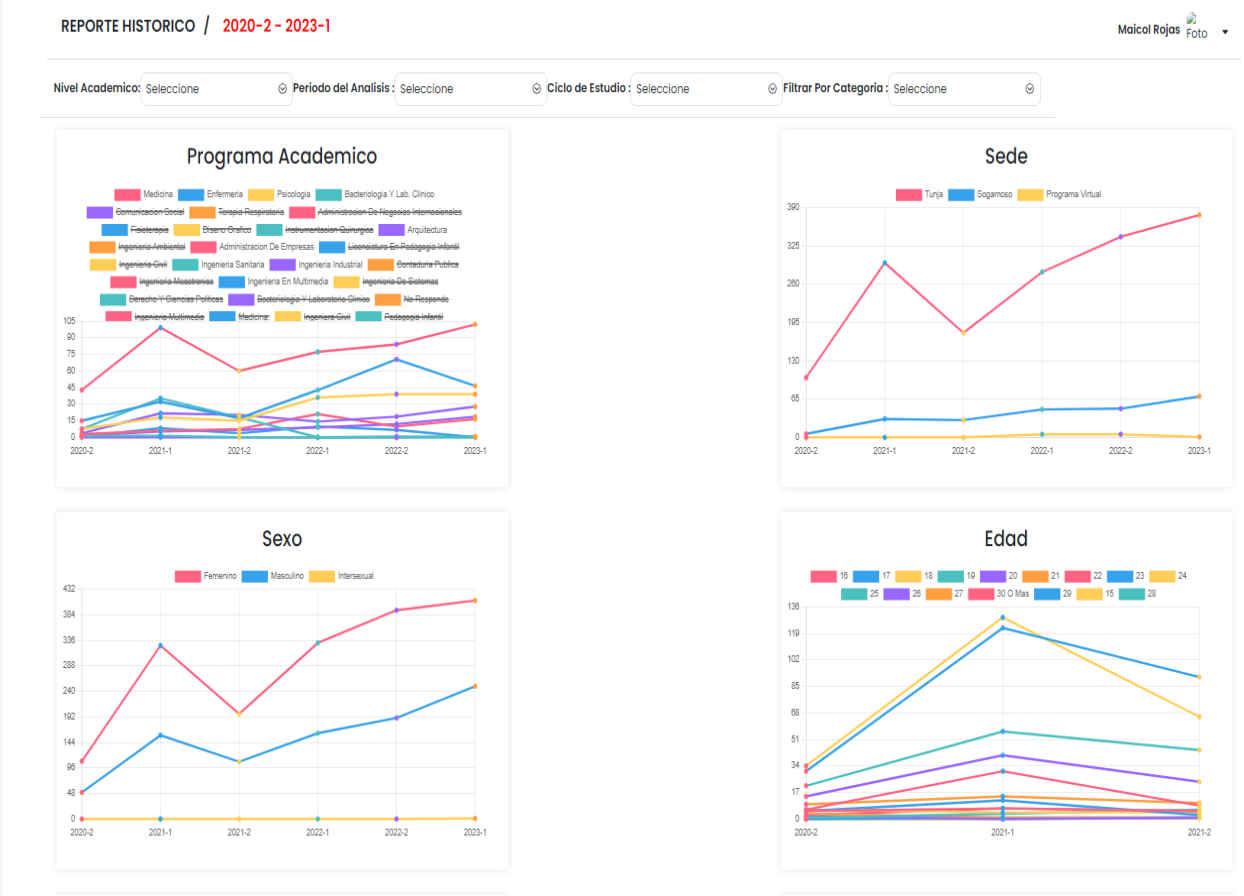
Fuente: elaboración propia

Reporte histórico

El módulo de reportes por periodo está diseñado para visualizar la información por un periodo histórico. Este contiene la página de reporte histórico, donde se aloja un encabezado, el menú lateral para facilitar la navegación dentro del módulo, y del mismo modo que el módulo de reportes por periodo se cuenta con la barra de herramientas que ayuda a filtrar los datos.

Figura 30

Vista general del módulo de reportes por periodo



Fuente: elaboración propia

Autenticación

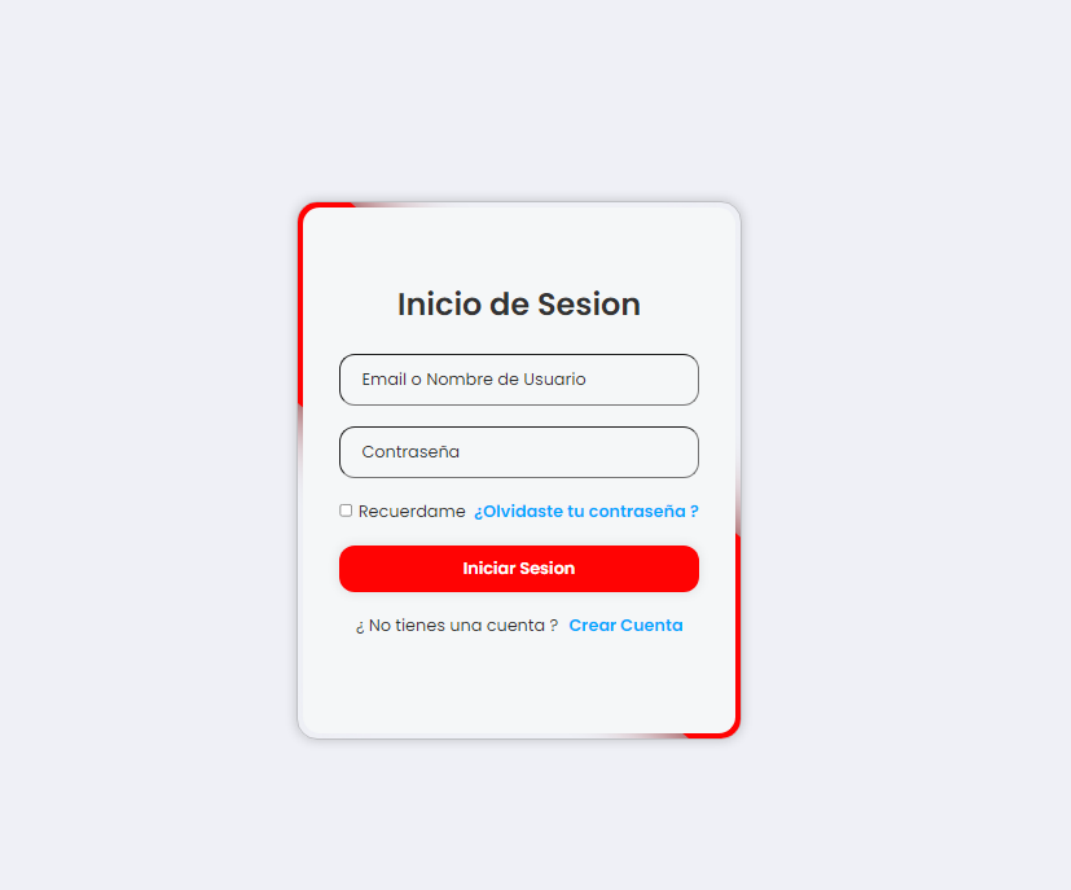
El módulo de autenticación se divide en dos secciones, el inicio de sesión actúa como un control de acceso. Permite a los usuarios autorizados ingresar al sitio proporcionando un nombre de usuario, correo electrónico y contraseña correspondientes. Esto se lleva a cabo a través de la validación de las credenciales ingresadas con la información almacenada en la base de datos. Solo si coinciden, se permite el acceso al usuario a las funciones y áreas protegidas del sitio.

Por otro lado, el registro de usuario permite como su nombre los indica registrar usuarios en la aplicación web. Este proceso permite a los nuevos usuarios crear cuentas en el sitio proporcionando información personal como nombre, dirección de correo electrónico y una

contraseña segura. Después de que esta información se envía y valida, se crea una nueva entrada en la base de datos del sistema, lo que permite al nuevo usuario iniciar sesión posteriormente.

Figura 31

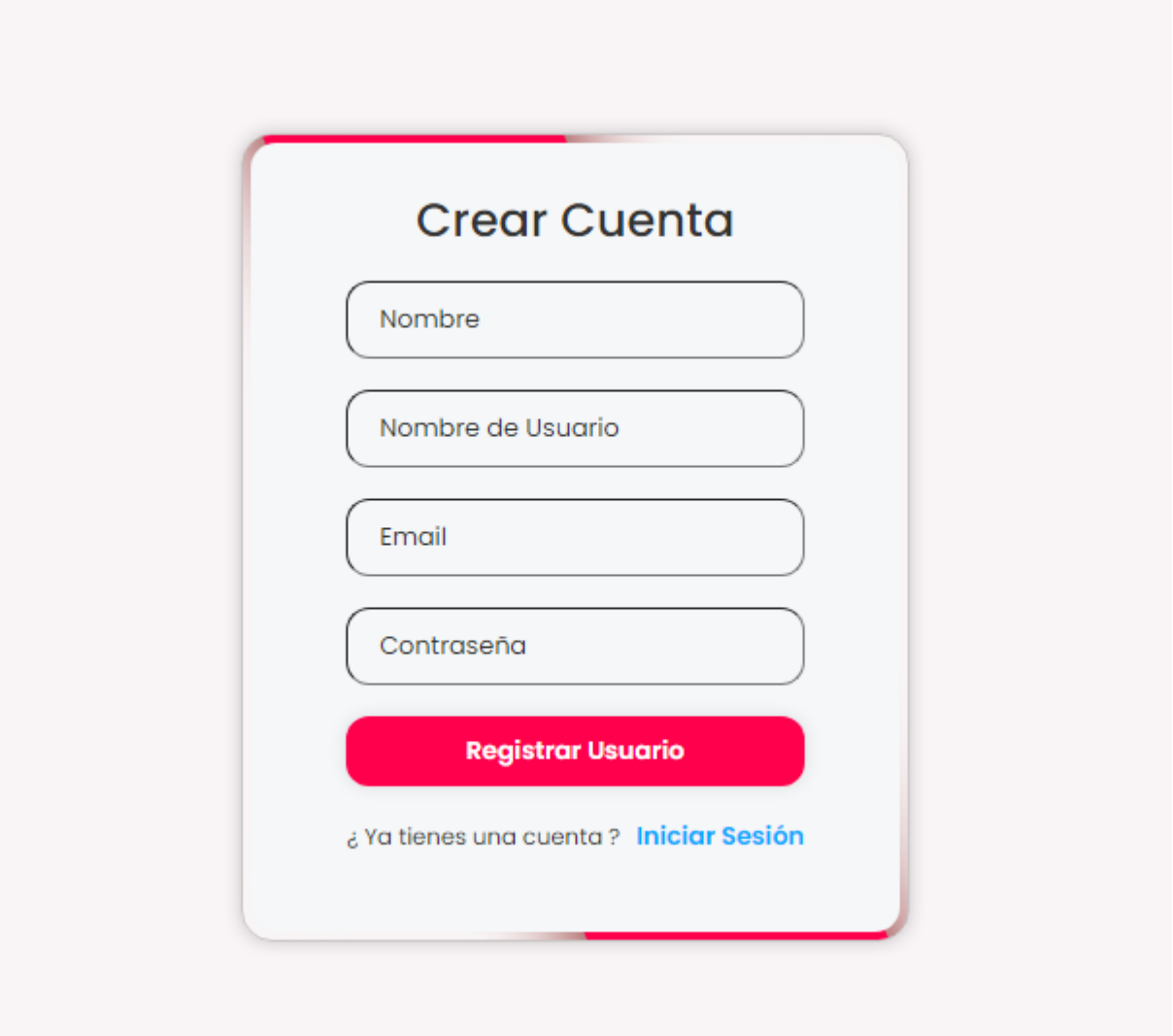
Vista general de la sección de inicio de sesión



Fuente: elaboración propia

Figura 32

Vista general de la sección de registro de usuario



Fuente: elaboración propia

Validación y despliegue del modelo

En este capítulo, se presentan los resultados correspondientes a las fases de validación del modelo construido y despliegue, en coherencia con la metodología CRISP-DM. Para esto se presentó la aplicación en funcionamiento a la directora del proyecto (exdirectora de DIPA) y a la actual directora de DIPA, con el fin de identificar si la aplicación cumplía con los requisitos previamente expresados, lo cual sirvió para realizar ajustes y poder realizar el despliegue de la aplicación que implementa el modelo de visualización de datos de caracterización de estudiantes nuevos, la cual, en este caso se realizó en un sitio web.

Evaluación

Presentación a los involucrados

Se realizó una exposición de los resultados alcanzados a los involucrados claves del proyecto (exdirectora y directora de DIPA). Durante esta presentación se procedió a explicar detalladamente el propósito del proyecto, la descripción del problema, las fuentes de los datos utilizadas, el modelo de datos aplicado y una explicación detallada de la funcionalidad de la aplicación web desarrollada. Así mismo, se proporcionó una descripción minuciosa del proceso llevado a cabo para llegar a esta etapa, desde la concepción del objetivo, la limpieza y modelado de los datos hasta el diseño e implementación de la aplicación, describiendo los problemas encontrados en los datos y demás obstáculos que se presentaron. Esta presentación permitió a los involucrados obtener una comprensión completa de la evolución y los logros del proyecto.

Realimentación

Una vez realizada la presentación del proyecto y de la aplicación web, la directora de DIPA (Ing. Sonia Milena Forero), manifestó sus observaciones al respecto y los aspectos por mejorar. La evidencia de la reunión sostenida, así como de la realimentación se puede ver en el Anexo E.

La realimentación realizada por parte de la Ingeniera Sonia Milena Forero fue:

- Se destacó el agrado por la funcionalidad del proyecto, debido a que éste brinda las funcionalidades establecidas y proporcionaba el valor extra requerido, gracias a su capacidad para visualizar los datos en diferentes periodos de tiempo y su habilidad para mostrar el nivel histórico de los datos.
- Se realizaron observaciones sobre la forma en que se mostraban algunas de las gráficas, sugiriendo la necesidad de modificar algunas para facilitar su entendimiento y análisis.
- Se sugirió buscar una mayor optimización el sitio web, debido a que en algunas consultas el tiempo de respuesta es elevado debido a las grandes cantidades de datos que debe procesar la aplicación.

.

Despliegue

Esta fase se centró en la puesta en marcha del software en un hosting. Para llevar a cabo esto, se hizo uso de la plataforma Railway, la cual ofrece el servicio de hosting gratuito. Para ello se conectó el repositorio alojado en GitHub a Railway, y posteriormente la plataforma detecto automáticamente los cambios del código base. Para el desarrollo de este proceso, se deben tomar en cuenta las configuraciones del entorno de desarrollo, dependencias necesarias y variables de entrono. Una vez finalizado el proceso anterior, Railway optimiza de manera automática las demás tareas como la creación del código, configuración del servidor y despliegue en la nube.

Inicialmente, se desplego el API de Django Rest Framework, como se puede observar en el siguiente enlace : https://web-production-7d5f.up.railway.app/data-explorer/api/v1/data/report_period?level=Pregrado&interval=Semestral&lapse=2023-1&category=&item=&item_data=.

Posteriormente, se realizaron los ajustes correspondientes al consumo del api desde el cliente, para una vez finalizada la actualización de las configuraciones se procedió a realizar el despliegue de la aplicación web en el siguiente host: https://web-production-7d5f.up.railway.app/.

Por otra parte, si la dependencia involucrada desea agregar datos de nuevos periodos académicos, puede hacerlo fácilmente mediante el código suministrado en el Anexo D. Para facilitar este proceso, se proporciona un manual de usuario en el Anexo F que detalla cómo agregar un nuevo periodo de manera sencilla.

Conclusiones

La metodología CRIPS-DM proporcionó una estructura metodología sólida y adecuada para la realización del proyecto de análisis de datos de manera exitosa, toda vez que facilitó el flujo procedimental para desarrollar el proyecto y permitió el logro de los objetivos fijados.

La fase de compresión del negocio presentada por la metodología CRISP-DM junto con el acompañamiento de la División de Planeación y Acreditación fueron piezas cables a la hora de formular los requisitos y objetivos, determinar los alcances y estipular las limitaciones del proyecto, de tal manera que gracias a esto se sentaron las bases del proyecto

El proceso de limpieza y depuración de datos a través de Python y la librería Pandas es crucial en un proyecto de analítica de datos, para garantizar que la información sea confiable y precisa.

El diseño del modelo de los datos implementado en la aplicación web construida, cumplió con las necesidades base planteadas en el desarrollo del proyecto y facilita el análisis de los datos de caracterización de estudiantes nuevos, a los interesados.

Python y su librería Pandas desempeñaron un papel fundamental en el proceso de análisis datos al facilitar la manipulación, limpieza y exploración de los subconjuntos de datos debido a la versatilidad de Python como lenguaje de programación y la utilidad de Pandas para la manipulación y transformación de los datos.

La implementación de las gráficas para la visualización de los datos utilizando ChartsJs simplificó de manera significativa el proceso de creación de estas, además de que el formato establecido es de fácil uso para la personalización de los resultados gráficos de los datos.

Durante el desarrollo del proyecto se enfrentaron dificultades considerables en la comprensión inicial y manejo de los datos debido a la diversidad de los formatos o tablas. Esta generó la necesidad de realizar múltiples validaciones para cada escenario, lo que dificultó la estandarización del producto final.

Recomendaciones

Con base en la experiencia adquirida durante el desarrollo del actual proyecto, se sugiere considerar las siguientes recomendaciones para futuras investigaciones que sigan una línea similar a la establecida en este proyecto.

Como primera recomendación para los interesados en trabajar proyectos de análisis de datos, se aconseja realizar un análisis profundo las herramientas y funcionalidades ofrecidas por lenguajes como Python y librerías como Pandas, esto como parte de la búsqueda de soluciones más reutilizables y escalables que algunas de las desarrolladas en este proyecto.

Del mismo modo se recomienda combinar proyectos de análisis de datos con proyectos relacionados con inteligencia artificial, con el fin de generar proyectos mucho más robustos combinando los posibles alcances de realizar un análisis con las proyecciones de datos que se pueden implementar por medio de la inteligencia artificial.

De igual manera se recomienda seguir con la implementación de otros proyectos en analítica de datos para la Universidad de Boyacá, en otras dependencias o en la del proyecto actual, ya que se ha demostrado que con una buena implementación y las bases suficientes se logra llevar a producción proyectos de gran utilidad para la mejora de los procesos administrativos de la Universidad.

Por último, es crucial destacar que, en proyectos como este, que involucran grandes conjuntos de datos de múltiples periodos de tiempo, es fundamental que las dependencias que generan estos datos cuenten con herramientas que validen los mismos, para evitar los problemas que constantemente se identifican en la calidad de los datos y que fueron mencionados en la fase de limpieza y transformación de los datos.

Referencias

Amazon Web Services. (s.f.). *Que es el analisis de datos*. https://aws.amazon.com/es/what-is/data-analytics/

Amazon Web Services. (s.f.). *¿Qué es una API? - Explicación de interfaz de programación de aplicaciones - AWS*. https://aws.amazon.com/es/what-is/api/#:~:text=API%20significa%20%E2%80%9Cinterfaz%20de%20programaci%C3%B3n,de%20servicio%20entre%20dos%20aplicaciones.

Arias, L. M. (2023, 08 de Noviembre). *Metodología CRISP-DM: La guía definitiva para la Minería de Datos*. https://www.linkedin.com/pulse/metodolog%C3%ADa-crisp-dm-la-gu%C3%ADa-definitiva-para-miner%C3%ADa-de-arias-xyusf/

Bustos, G. (2024, 25 de Enero). *¿Qué es un hosting y cómo funciona?*. https://www.hostinger.co/tutoriales/que-es-un-hosting

Casas Roma, J., Nin Guerrero, J. y Julbe López, F. (2019). *Big data: análisis de datos en entornos masivos.* Editorial UOC.

Fernandez, R., Roca, J., Costa, J. y Oviedo, M. (2023). *Data frames | Introducción al Análisis de Datos con R*. https://rubenfcasal.github.io/intror/data-frames.html

Hinajosa Gutierrez, A. (2015). *Python paso a paso.* RA-MA Editorial.

IBM. (s.f.). *Analisis Exploratorio de Datos*. https://www.ibm.com/mx-es/topics/exploratory-data-analysis

JSON. (s.f.). *Json*. https://www.json.org/json-es.html

MDN Web Docs. (s.f.). *¿Qué es una URL? - Aprende desarrollo web*. https://developer.mozilla.org/es/docs/Learn/Common_questions/Web_mechanics/What_is_a_URL

MDN Web Docs. (s.f.). *HTTP*. https://developer.mozilla.org/es/docs/Web/HTTP

MDN Web Docs. (s.f.). *¿Qué es JavaScript? - Aprende desarrollo web*. https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/What_is_JavaScript

Pollo Cattaneo, M. F. (2018). *Modelo de proceso para la elicitación de requerimientos en proyectos de explotación de información.* (Tesis de maestría, Universidad Nacional de La Plata). SEDICI - Repositorio institucional de la UNLP. http://sedici.unlp.edu.ar/handle/10915/66760

Railway. (s.f.). *About Railway | Railway Docs*. https://docs.railway.app/overview/about-railway

React. (s.f). *Describir la UI*. https://es.react.dev/learn/describing-the-ui