

DOCUMENTACIÓN PROYECTO FINAL.

Python para el análisis de datos



Álvaro José Assis Arciria	CC 1065007846
Carlos Andrés tajan Ruiz	CC 1003398770
Fabricio Javier Durango Falon	CC 1007734204
Jonathan Esteban castro padilla	CC 1067926352
Marcos David García negrete	CC 1067961966

Universidad de Córdoba
Facultad de Ingeniería
Departamento de Ingeniería de Sistemas y Telecomunicaciones
Ingeniería de Sistemas
Diciembre 17 - 2022
Montería

CONTENIDO.

1. JUSTIFICACION.....	3
2. DESCRIPCION FUNCIONAL DEL SISTEMA.....	4
3. METODOLOGIA DE DESARROLLO.....	6
3.1. Planificación del sprint.....	7
3.2. Planificación diría del proyecto.	7
3.3. Etapa de desarrollo.....	7
3.4. Revisión del sprint.....	8
3.5. Retroalimentación.....	8
4. ARQUITECTURA DEL SISTEMA.....	9
5. ANALISIS Y DISEÑO DEL SISTEMA.....	10
5.1. Preprocesamiento De Los Datos.....	16
5.2. Modelamiento.....	17
6. ESPECIFICACIONES DE REQUISITOS FUNCIONALES / NO FUNCIONALES.....	19
6.1. Requisitos Funcionales.....	19
6.2. Requisitos No Funcionales.....	19
7. OBJETIVOS DEL SISTEMA.....	20
7.1. Objetivo General.	20
7.2. Objetivos Específicos.....	20
8. UML.....	21
8.1. Casos de Uso:	21
8.2. Diagrama de Actividades:	23
8.3. Diagrama Modelo Entidad-Relación:.....	24
8.4. Diagrama de Componentes:.....	25
9. ANEXO.....	26
9.1. MANUAL DE USUARIO.....	26
REFERENCIAS.....	30

1. JUSTIFICACION.

Python es un lenguaje de programación multiplataforma de código abierto, el cual soporta parcialmente la orientación a objetos, programación imperativa y la programación funcional, lo cual nos permite crear todo tipo de aplicaciones. Gracias a que es un lenguaje sencillo de leer y escribir, aspecto que lo ha hecho ganar adeptos y posibilidades de expansión como en la inteligencia artificial, big data, machine Learning y data science.

Ha tenido un impacto impresionante, tanto así que las empresas han empezado a implementar este lenguaje en sus aplicaciones ya que es muy efectivo y rápido. Permitiendo así tener mayor productividad y rendimiento en sus tareas. En la actualidad nos vemos cada día más sumergidos en la famosa Inteligencia Artificial, Blockchain y el Machine Learning. Por tal razón nos vemos obligados según la demanda que hay en el mundo laboral a aprender este lenguaje, que es y será de mucha relevancia en la actualidad y en el futuro.

Python dispone de un intérprete por línea de comandos en el que se pueden introducir sentencias. Cada sentencia se ejecuta y produce un resultado visible, que puede ayudarnos. Machine Learning es una disciplina científica del ámbito de la Inteligencia Artificial que crea sistemas que aprenden automáticamente. Aprender en este contexto quiere decir identificar patrones complejos en millones de datos. Es una tecnología que permite hacer automáticas una serie de operaciones con el fin de reducir la necesidad de que intervengan los seres humanos. Esto puede suponer una gran ventaja a la hora de controlar una ingente cantidad de información de un modo mucho más efectivo.

El lenguaje Python juega un papel fundamental en el contexto de la analítica de datos y Big Data, ya que dispone de las herramientas para casi todos los aspectos relacionados con la computación científica. Hasta ahora, entre los lenguajes que más se usaban para realizar análisis y visualización de datos se encontraban Matlab y R. Sin embargo, en los últimos años Python se ha hecho muy popular entre los desarrolladores de aplicaciones y analistas de datos.

2. DESCRIPCION FUNCIONAL DEL SISTEMA.

EL aplicativo web creado como proyecto final para el diplomado “Python para el análisis de datos”, consta de dos partes principales, la barra lateral y el cuerpo principal

-Barra lateral

En esta podemos allá todos los selectores destinados para que el usuario final interactúe con el dashboard, en ella se encuentran selectores del tipo slider y selectbox, que interactúan de forma directa con la predicción que crea el api por fuera del dashboard, y también con las posibles graficas que puede manipular el usuario haciendo uso de los selectores.

-Cuerpo principal

Dentro del contenedor principal se encuentran los visualizadores de datos, en él se encuentran las predicciones, los datos utilizados para modelo de Maching Learning,

las gráficas interactivas, y un buscador que contiene información de la plataforma Cintia de la Universidad de Córdoba,

En el primer visualizador se encuentra la predicción traída directamente desde el api web, luego tenemos la vista de todos los datos en forma tabla, donde se pueden todos los datos utilizados para predecir , y estas son todas las funcionalidades que componen el dashboard.

3. METODOLOGIA DE DESARROLLO

Las metodologías ágiles son un tema candente y más escuchado en ingeniería de software que están acaparando de una forma muy contundente. Prueba de ello es que se están haciendo un espacio destacado en la mayoría de conferencias y workshops celebrados en los últimos años. Además, ya es un área con cabida en prestigiosas revistas internacionales. En la comunidad de la ingeniería del software, se está viviendo con intensidad un debate abierto entre los partidarios de las metodologías tradicionales (referidas peyorativamente como "metodologías pesadas") y aquellos que apoyan las ideas emanadas del "Manifiesto Ágil".

Las metodologías ágiles están especialmente orientadas para proyectos pequeños, estas constituyen una solución a medida para ese entorno, aportando una elevada simplificación que a pesar de ello no renuncia a las prácticas esenciales para asegurar la calidad del producto. (Penadés & Torres, 2006).

Metodología Scrum: Esta metodología es incremental, donde tenemos etapas para realizar el trabajo (análisis, desarrollo y testing). Y es aquí donde planificamos brevemente como realizar el trabajo, cuantos ciclos y el tiempo que durará. Resulta imprescindible el empleo de metodologías ágiles para el desarrollo del aplicativo propuesto. Ya que se adapta a la realidad tecnológica y su constante evolución, admitiendo desarrollar la app en periodos cortos de entre uno y seis meses, realizando varias actualizaciones según se vayan entregando nuevas funcionalidades (Martin, 2019) .

En base a esto se utilizará esta metodología ágil para el desarrollo de este proyecto ya que cuenta con una mayor rapidez y adaptabilidad de resultados al momento de realizar proyectos de corto plazo. Scrum posee 5 etapas o fases fundamentales, las cuales están definidas y dirigidas al desarrollo de este proyecto y las cuales son:

3.1. Planificación del sprint.

Esta primera fase se definirá aspectos como la funcionalidad, objetivos, riesgos, plazos de entrega, entre otros, los cuales son fundamentales para el desarrollo del proyecto. Posteriormente se realiza una junta entre el equipo para explicar cómo se desarrollará cada punto del intervalo. Aquí se evaluarán cambios, toma de decisiones, mejoras y más factores.

3.2. Planificación diaria del proyecto.

En esta segunda fase o etapa se planificará durante 15 minutos al comienzo de la jornada, las actividades establecidas en el proyecto y se creará un plan de trabajo durante las próximas horas de la jornada. El equipo de desarrollo utilizará la planificación diaria para evaluar el progreso hacia la meta del Sprint y evaluar la tendencia del progreso en finalizar el trabajo.

3.3. Etapa de desarrollo.

En esta fase cuando el trabajo del sprint está en curso de desarrollo, los encargados garantizarán que no se generen cambios de último momento que puedan

afectar los objetivos de este proyecto, tales como requerimientos a última hora o cambios sobre la marcha. Además, se asegurará el cumplimiento de los plazos establecidos para el termino satisfactorio del proyecto.

3.4. Revisión del sprint.

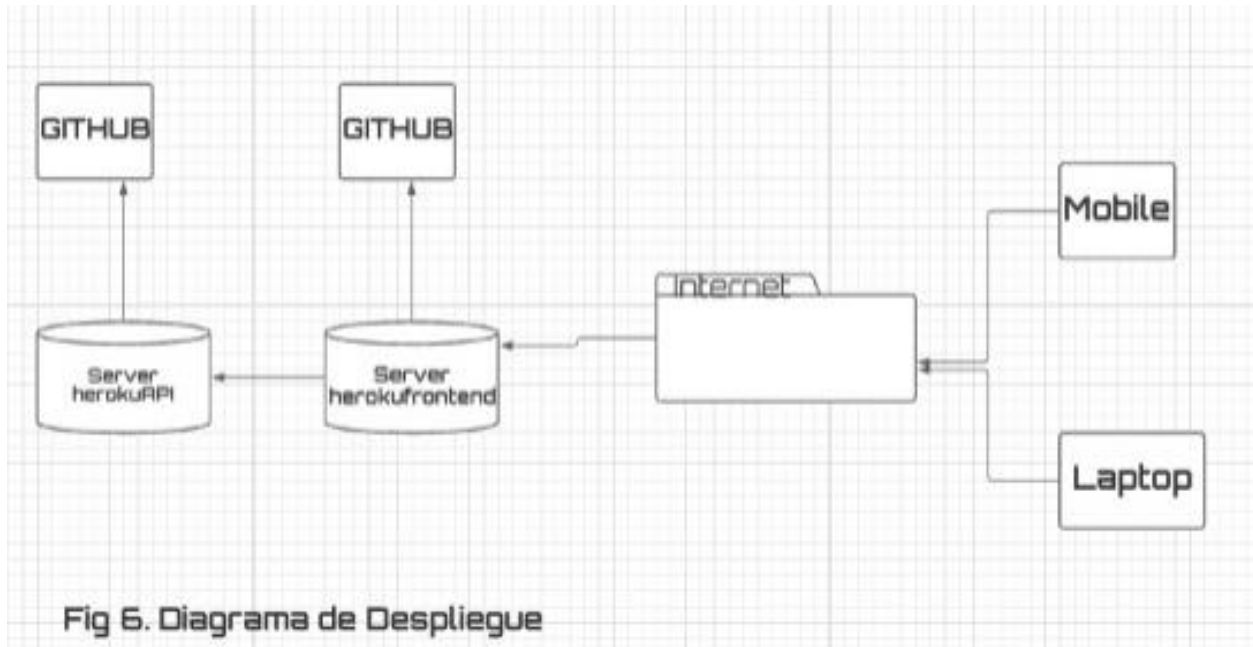
En esta fase al final del desarrollo estipulado, se hará posible analizar y evaluar los resultados y verificar que cumplen con todos los requerimientos. Si es necesario, el equipo colaborará para saber qué aspectos necesitan ser cambiados.

3.5. Retroalimentación.

En esta fase o etapa los resultados que serán entregados pueden recibir un feedback no solo por parte de los profesionales dentro del proyecto, sino también de las personas que utilizarán directamente el aplicativo. Las lecciones aprendidas durante esta etapa permitirán que el siguiente sprint pueda ser mucho más efectivo y ágil.

Con esta metodología se obtiene un trabajo más dinámico, claro, con un aumento de la productividad y la calidad del producto y este es el factor que las empresas persiguen hoy día, que su producto sea flexible, autónomo, eficaz y que implique bajo costo de producción pero que genere gran impacto en el negocio.

4. ARQUITECTURA DEL SISTEMA.

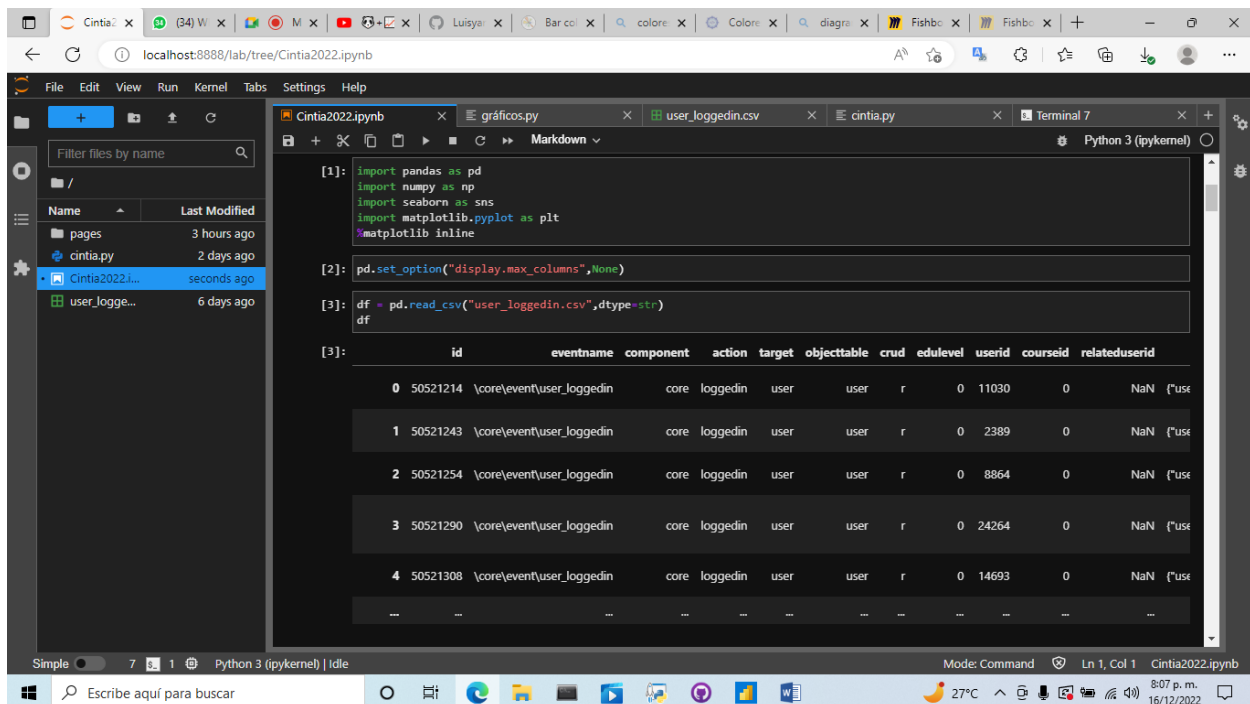


Para la arquitectura y despliegue de este aplicativo se tuvo en cuenta trabajar con dos repositorios en GitHub uno donde se encuentra alojado la API la cual está conectada al modelo de predicción, así como el mismo modelo y otro repositorio donde se encuentra alojado el Dashboard. Tanto la API como el Dashboard fueron subidos a los servidores de GitHub y Streamlit por donde ya se pueden acceder a ellos por medio de celulares o laptops/pc siempre que se tenga el link de acceso.

5. ANALISIS Y DISEÑO DEL SISTEMA.

En este segmento, se hablará sobre el análisis de los datos encontrados en el dataset con el nombre `user_loggedin.csv` correspondiente a nuestro grupo, que, según la investigación realizada, el dataset proviene de una plataforma llamada CINTIA la cual es una plataforma de la Universidad de Córdoba ubicada en la ciudad de Montería, por ende, podemos asumir que el dataset nos mostrara datos con respecto a los estudiantes de la Universidad para así ilustrar o visionar un modelo preliminar de Machine Learning para predecir lo que hallemos en este análisis.

En este caso lo primero que haremos será cargar los datos.



The screenshot shows a Jupyter Notebook environment with the following components:

- File Explorer (Left):** Lists files including `pages`, `cintia.py`, `Cintia2022...`, and `user_logge...`.
- Code Editor (Center):** Contains the following Python code:

```
[1]: import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline

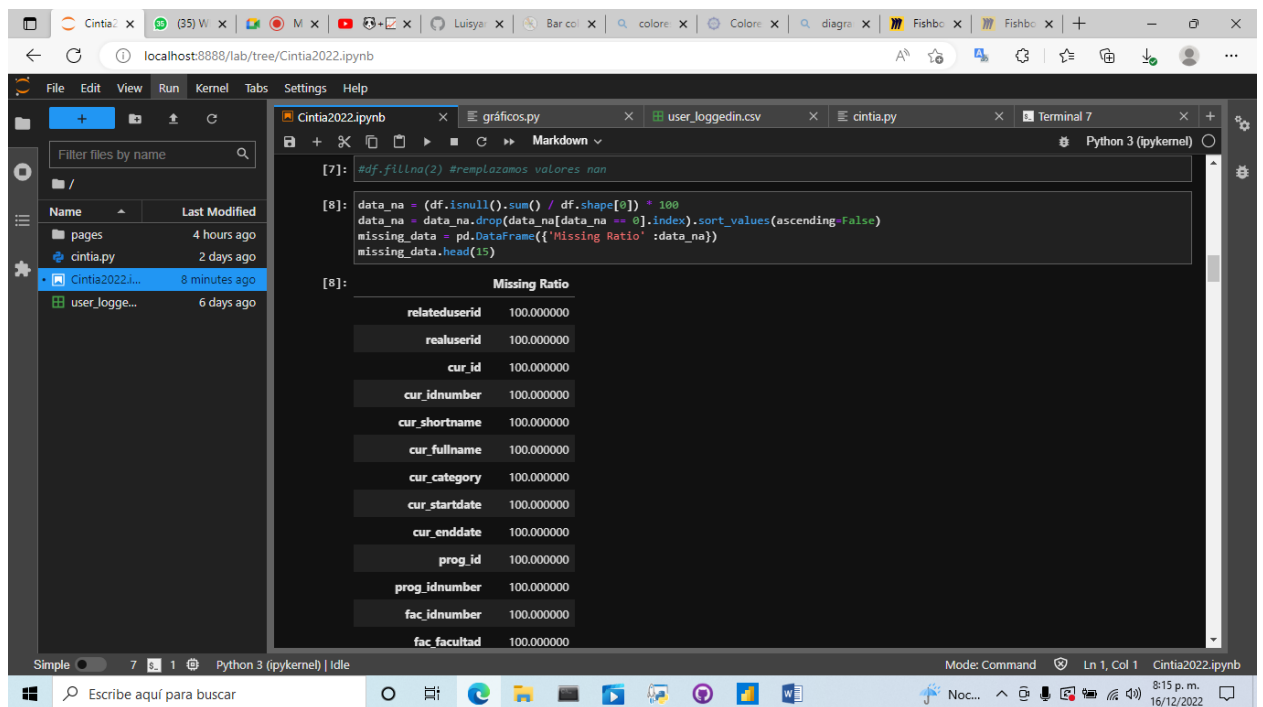
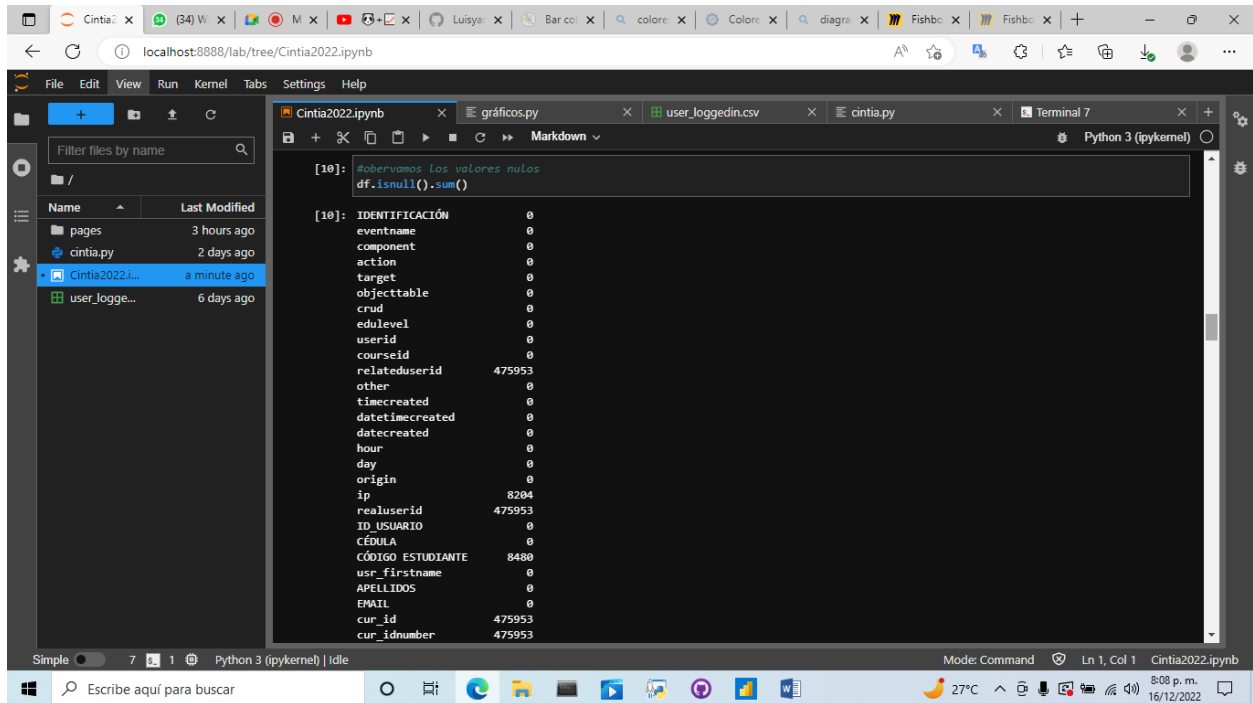
[2]: pd.set_option("display.max_columns",None)

[3]: df = pd.read_csv("user_loggedin.csv",dtype=str)
df
```
- Output (Right):** Displays the first five rows of the loaded DataFrame:

	id	eventname	component	action	target	objecttable	crud	edulevel	userid	courseid	relateduserid
0	50521214	\core\event\user_loggedin	core	loggedin	user	user	r	0	11030	0	NaN
1	50521243	\core\event\user_loggedin	core	loggedin	user	user	r	0	2389	0	NaN
2	50521254	\core\event\user_loggedin	core	loggedin	user	user	r	0	8864	0	NaN
3	50521290	\core\event\user_loggedin	core	loggedin	user	user	r	0	24264	0	NaN
4	50521308	\core\event\user_loggedin	core	loggedin	user	user	r	0	14693	0	NaN

The status bar at the bottom indicates the notebook is running on Python 3 (ipykernel) and shows the current mode as Command.

Fig 1. Procedemos a verificar si en el dataset se encuentran presentes datos NaN.



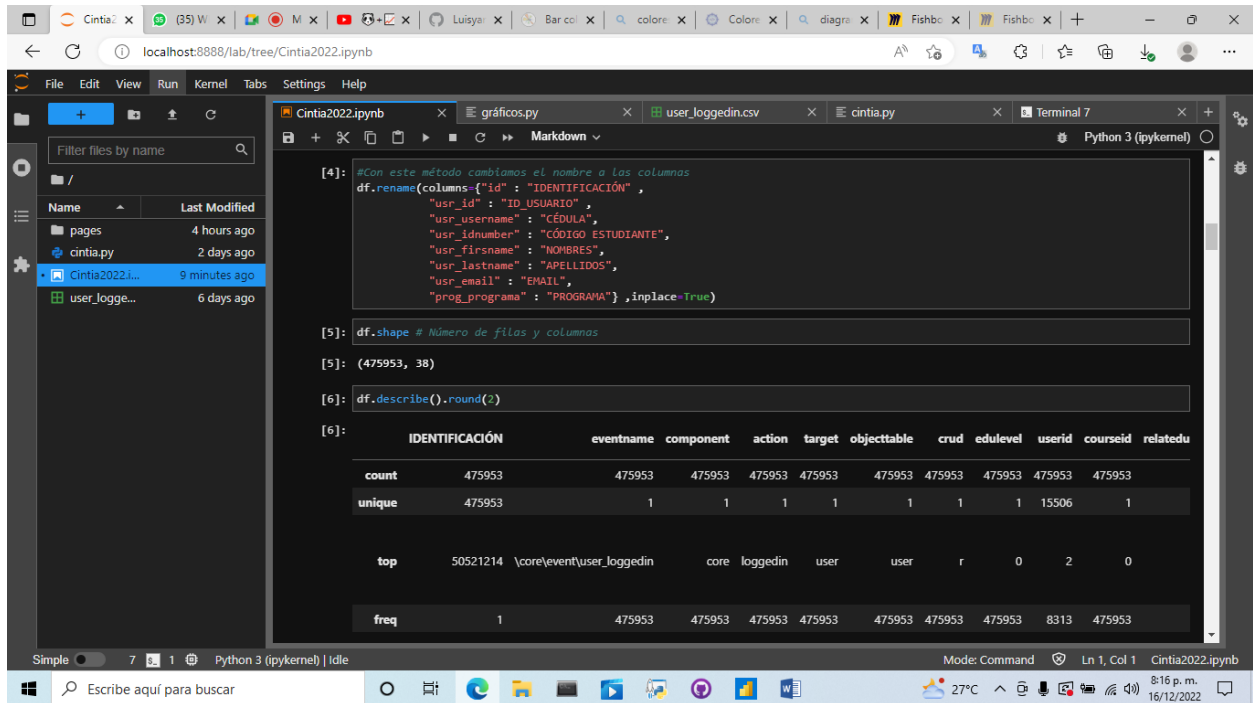


Fig 4. Renombramos algunas columnas, como vemos en la figura

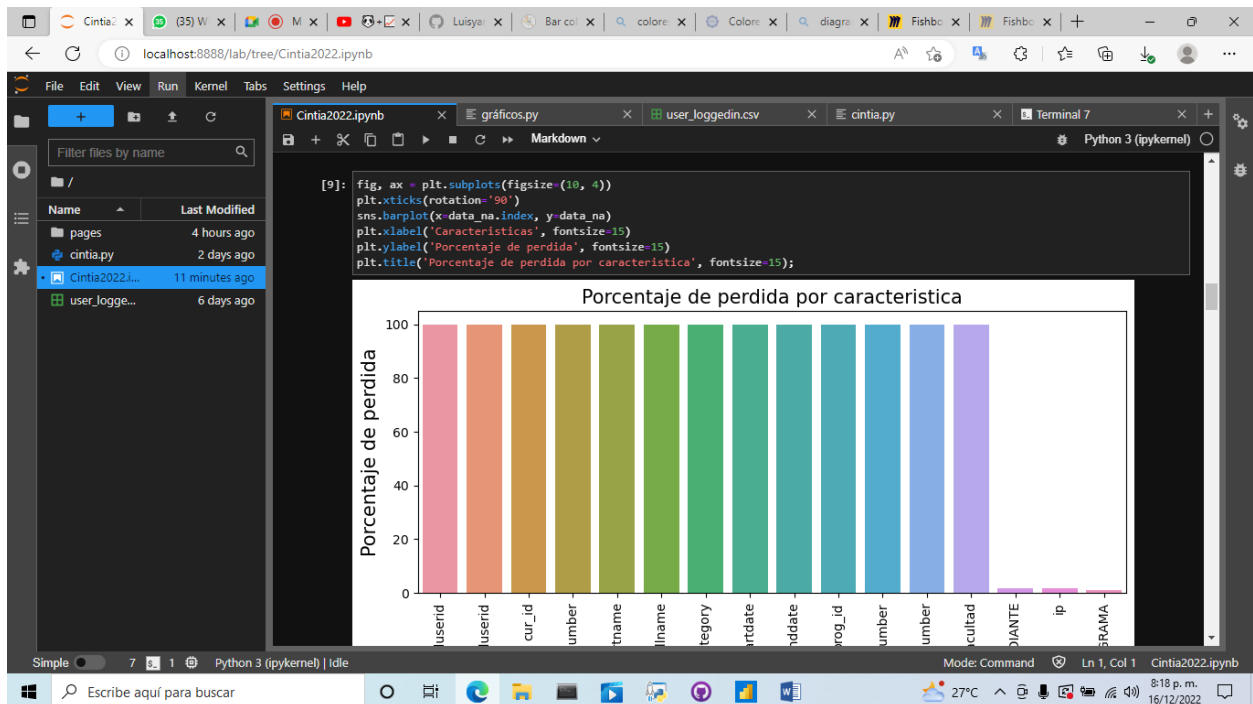


Fig 5. Se procede a visualizar el porcentaje de pérdida

```
[18]: df.columns

[18]: Index(['IDENTIFICACIÓN', 'EVENTNAME', 'COMPONENT', 'ACTION', 'TARGET',
          'OBJECTTABLE', 'CRUD', 'EDULEVEL', 'USERID', 'COURSEID', 'OTHER',
          'TIMECREATED', 'DATETIMECREATED', 'DATECREATED', 'HOUR', 'DAY',
          'ORIGIN', 'IP', 'ID_USUARIO', 'CÉDULA', 'CÓDIGO ESTUDIANTE',
          'USR_FIRSTNAME', 'APELLIDOS', 'EMAIL', 'PROGRAMA'],
          dtype='object')
```

Fig. 6. Observamos las columnas de nuestro Dataset

```
[14]: df["PROGRAMA"].value_counts()

[14]: ADMINIS. EN FINANZAS Y NEGOCIOS INTERNAC    80563
      ADMINISTRACIÓN EN SALUD                49156
      INGENIERÍA DE SISTEMAS                 47583
      INGENIERÍA INDUSTRIAL                 26707
      LICENCIATURA EN INFORMATICA           20903
      ...
      INGENIERÍA AGRONOMIA                   1
      DIVISION DE POSTGRADOS Y EDUCACION CONTINUADA 1
      FUNCIONARIOS ADM. PLANTA               1
      DECANATURA/GESTOR DE CALIDAD           1
      DPTO DE GEOGRAFIA Y MEDIO AMBIENTE      1
      Name: PROGRAMA, Length: 124, dtype: int64
```

Fig. 7. Observamos la columna programas con su respectiva cantidad de alumnos que pertenecen a dicho programa.

```
[20]: df.groupby("DAY")["ID_USUARIO"].count( )

[20]: DAY
      DOMINGO    48530
      JUEVES    72999
      LUNES     52804
      MARTES    78740
      MIERCOLES  82092
      SABADO    67779
      VIERNES   73009
      Name: ID_USUARIO, dtype: int64
```

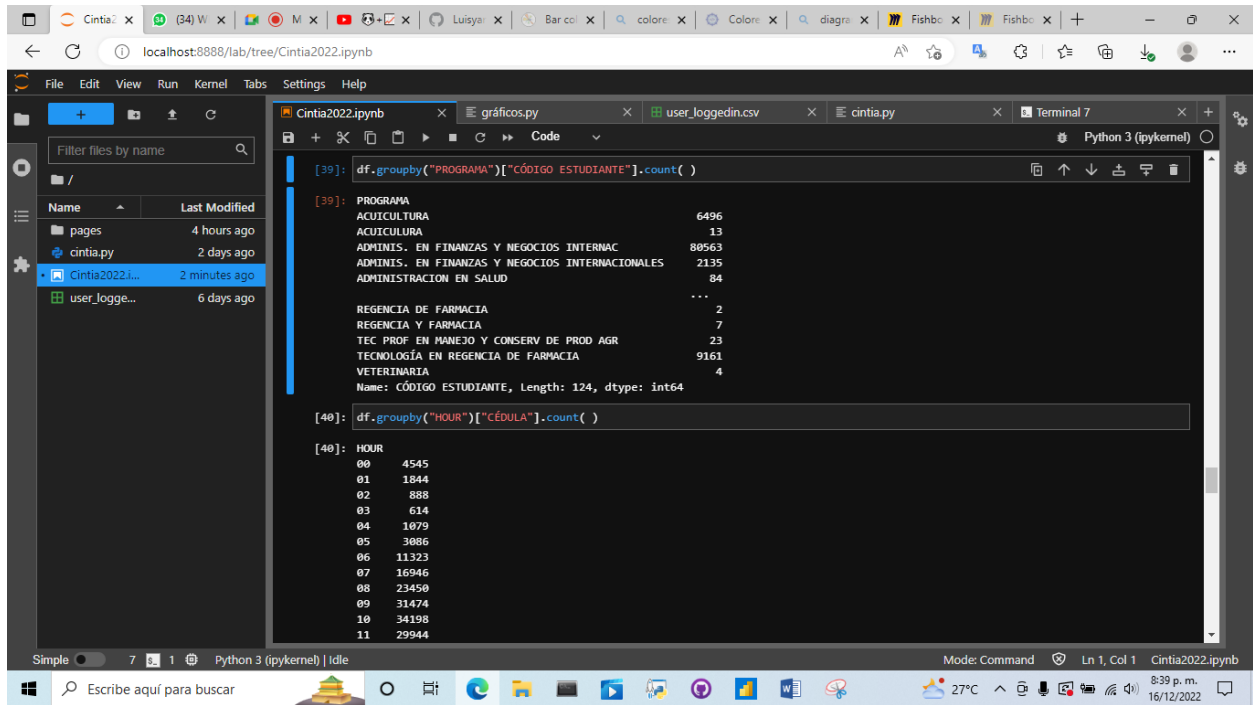


Fig 8. Con el método groupby agrupamos las dos columnas para ver la información detallada

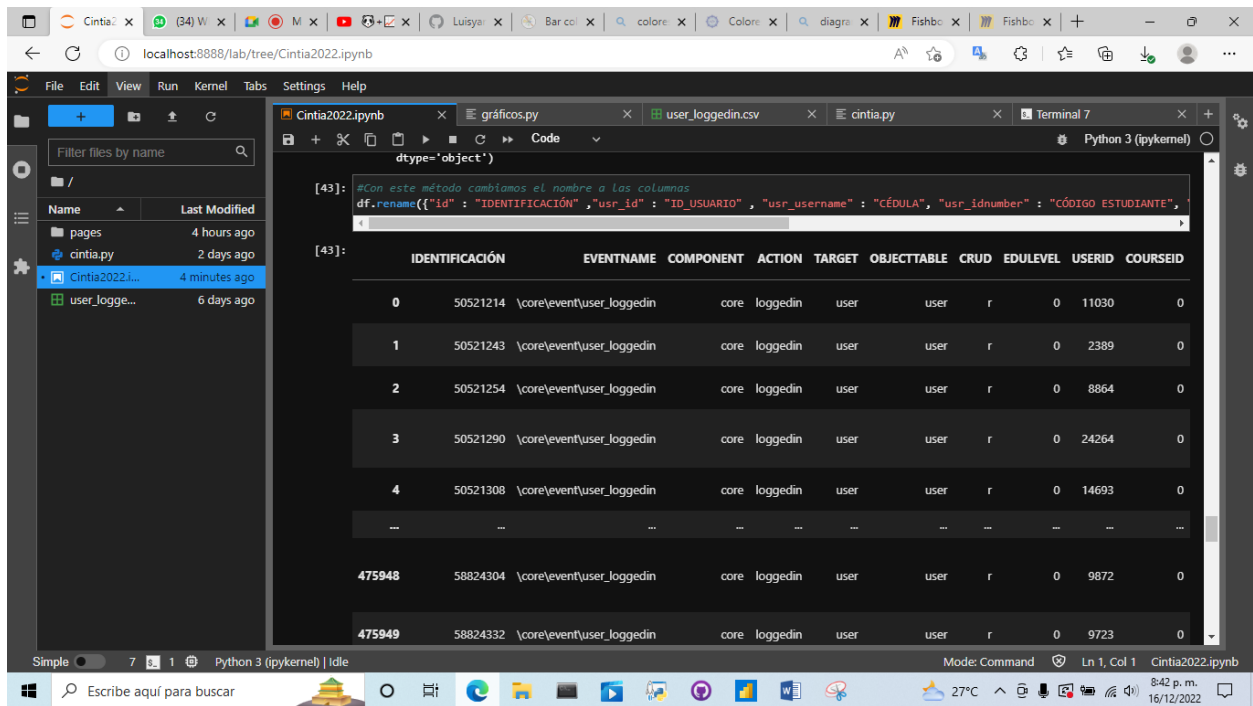


Fig 9. Con el método rename renombramos algunas columnas.

Teniendo en cuenta el análisis anterior profundizaremos en algunas columnas para verificar la lógica que tiene el dataset. Se revisará a mayor profundidad los datos para detectar ciertas interacciones y guiar la futura interpretación de los modelos.

Para ello, se revisarán interacciones, agrupaciones y visualizaciones de los datos, y se generarán ideas de las algunas interacciones interesantes que se detecten.

- ❖ Al agrupar los datos por alguna de sus variables, se observa que alguna otra tiene un nivel diferente.
- ❖ Generar algunas gráficas para apoyar las posibles observaciones.

```
df.groupby('PROGRAMA').mean()
```

	IDENTIFICACIÓN	EDULEVEL	USERID	COURSEID	TIMECREATED	HOUR	ID_USUARIO
PROGRAMA							
ACUICULTURA	inf	0.0	inf	0.0	inf	inf	inf
ACUICULTURA	3.975365e+102	0.0	1.736787e+63	0.0	1.278184e+128	1.323217e+24	1.736787e+63
ADMINIS. EN FINANZAS Y NEGOCIOS INTERNAC	inf	0.0	inf	0.0	inf	inf	inf
ADMINIS. EN FINANZAS Y NEGOCIOS INTERNACIONALES	inf	0.0	inf	0.0	inf	inf	inf
ADMINISTRACION EN SALUD	inf	0.0	inf	0.0	inf	2.166787e+165	inf
...
REGENCIA DE FARMACIA	2.529107e+15	0.0	8.831088e+08	0.0	8.305045e+18	5.050000e+02	8.831088e+08
REGENCIA Y FARMACIA	7.226041e+54	0.0	2.731599e+33	0.0	2.372898e+68	2.164545e+12	2.731599e+33
TEC PROF EN MANEJO Y CONSERV DE PROD AGR	2.244710e+182	0.0	6.931808e+112	0.0	7.223977e+227	2.639530e+43	6.931808e+112
TECNOLOGÍA EN REGENCIA DE FARMACIA	inf	0.0	inf	0.0	inf	inf	inf
VETERINARIA	1.370972e+31	0.0	7.660766e+14	0.0	4.161405e+38	3.272752e+06	7.660766e+14

124 rows × 7 columns

Fig 10. Se observa que los promedios de los estudiantes pertenecientes a algún programa en específico

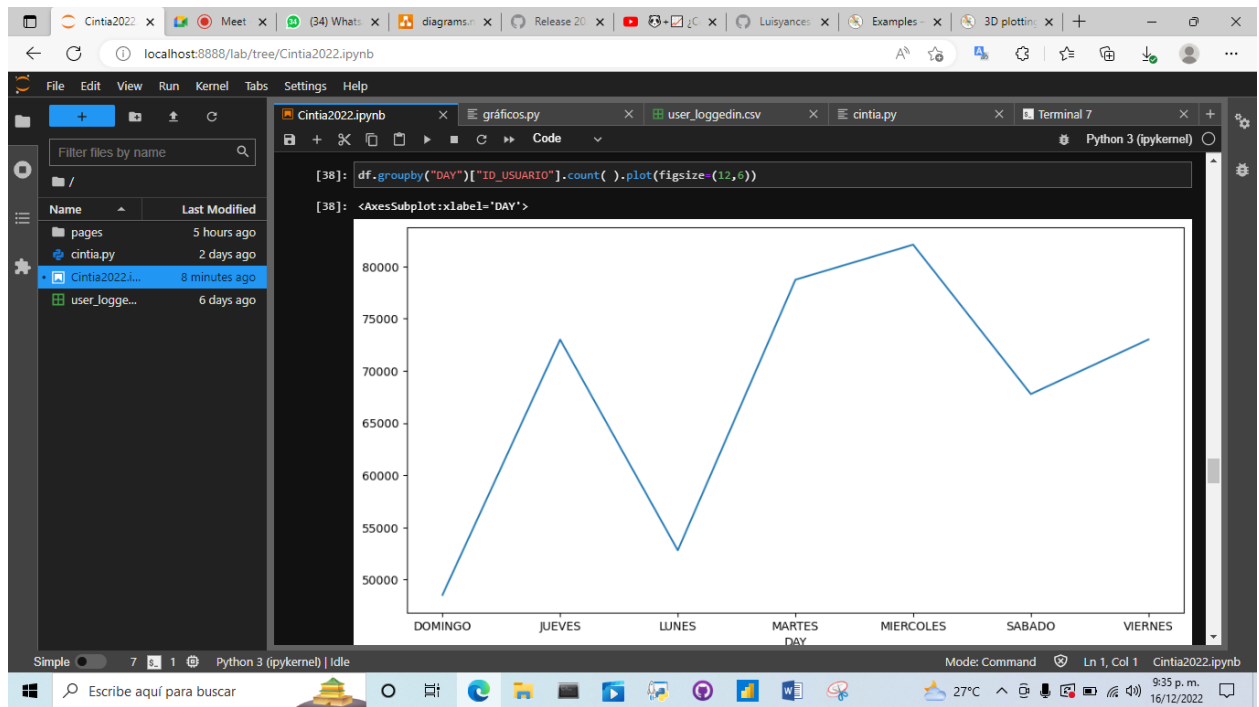


Fig 11. Se comparan el día de la semana y la cantidad de usuarios que se loguean

5.1. Preprocesamiento De Los Datos.

Se aplica cambios a los datos originales, donde tenemos tanto variables tanto numéricas como categorías donde se probarán inicialmente los modelos:

❖ RandomForestClassifier

Teniendo en cuenta lo descubierto en el análisis exploratorio, partiremos el dataset para sacar las columnas que requerimos.

5.2. Modelamiento.

Se usarán funciones para partir los datos y usar Onehot para los datos categóricos.

Sidebar

```
# Sidebar
st.sidebar.image("https://upload.wikimedia.org/wikipedia/commons/thumb/c/c3/Python-logo-notext.svg/800px-Python-logo-notext.svg.png")
st.sidebar.markdown("# Selectores de datos para estimador de id")
st.sidebar.markdown("---")
id = st.sidebar.slider(
    label = "id", min_value=50521214, max_value=58824449)
Potencia = st.sidebar.slider(
    label="usr_id", min_value=40, max_value=24984, value=475953
)
Clase = st.sidebar.selectbox(
    label="prog_programa", options=["INGENIERIA AMBIENTAL", "Licenciatura En Educación Infantil", "INGENIERIA MECÁNICA",
    "QUÍMICA", "INGENIERIA DE ALIMENTOS", "ADMINISTRACIÓN EN SALUD", "ACUICULTURA", "LIC EN LITERATURA Y LENGUA CASTELLANA",
    "Ingeniería de Alimentos", "Matemáticas", "Adminis. en Finanzas y Negocios Internac", "Física", "Medicina Veterinaria y Zootecnia",
    "GEOGRAFÍA", "INGENIERÍA DE SISTEMAS", "Ingeniería Agronómica", "LIC EN LENGUAS EXTRAN CON ENFA EN INGLES", "Lic en Ciencias Naturales y Edu Ambien", "BACTERIOLOGÍA", "ESTADÍSTICA", "LICENCIATURA EN INFORMATICA", "Derecho", "BIOLOGÍA"])
```

Fig 12. Opciones del sidebar

Aplicaremos estas funciones a los datos y usamos el primer modelo establecido que es el RandomForestRegressor.

```
1 from pydantic import BaseModel as PydanticBaseModel
2 from pydantic import Field, ValidationError
3 from typing import Literal
4 import joblib
5 import pandas as pd
6 import datetime as dt
7 import os
8 import numpy as np
9 from fastapi import HTTPException
10 |
11 class ModelInput(PydanticBaseModel):
12     """
13     Clase que define las entradas del modelo
14     """
15
```

Fig 13. Observamos las librerías que usamos en nuestro DataFrame

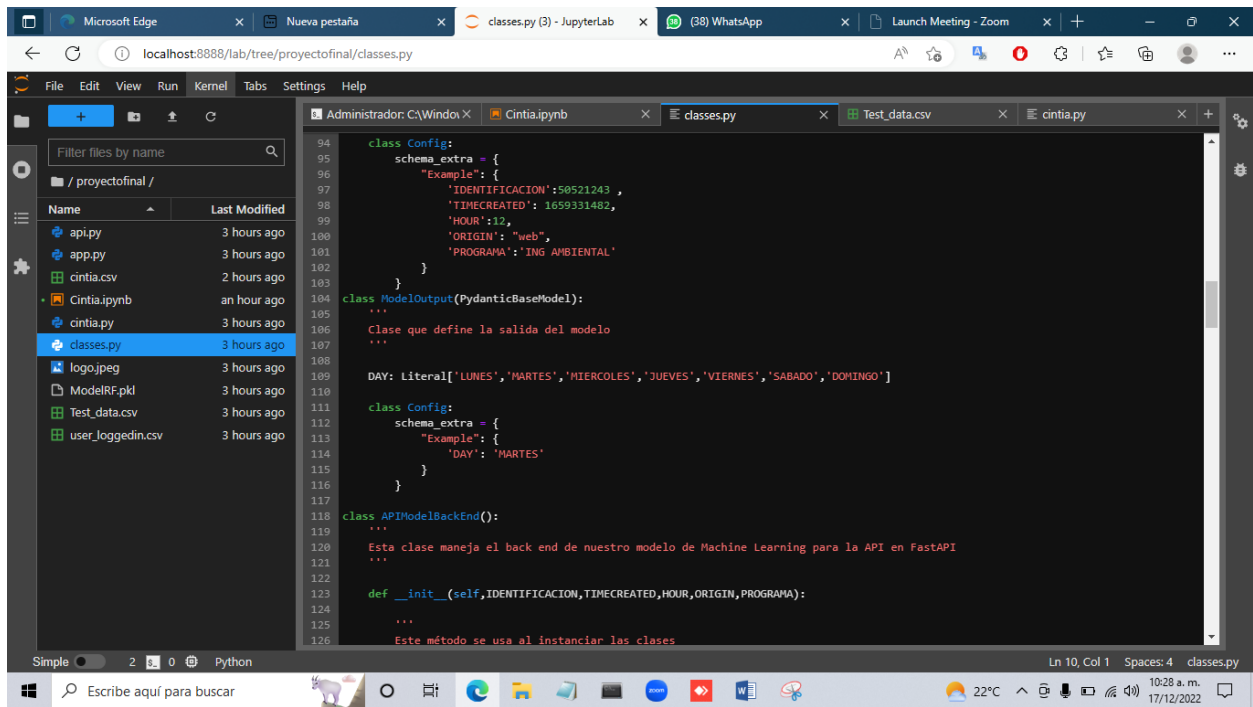


Fig 14. Observamos las Clases para el modelo

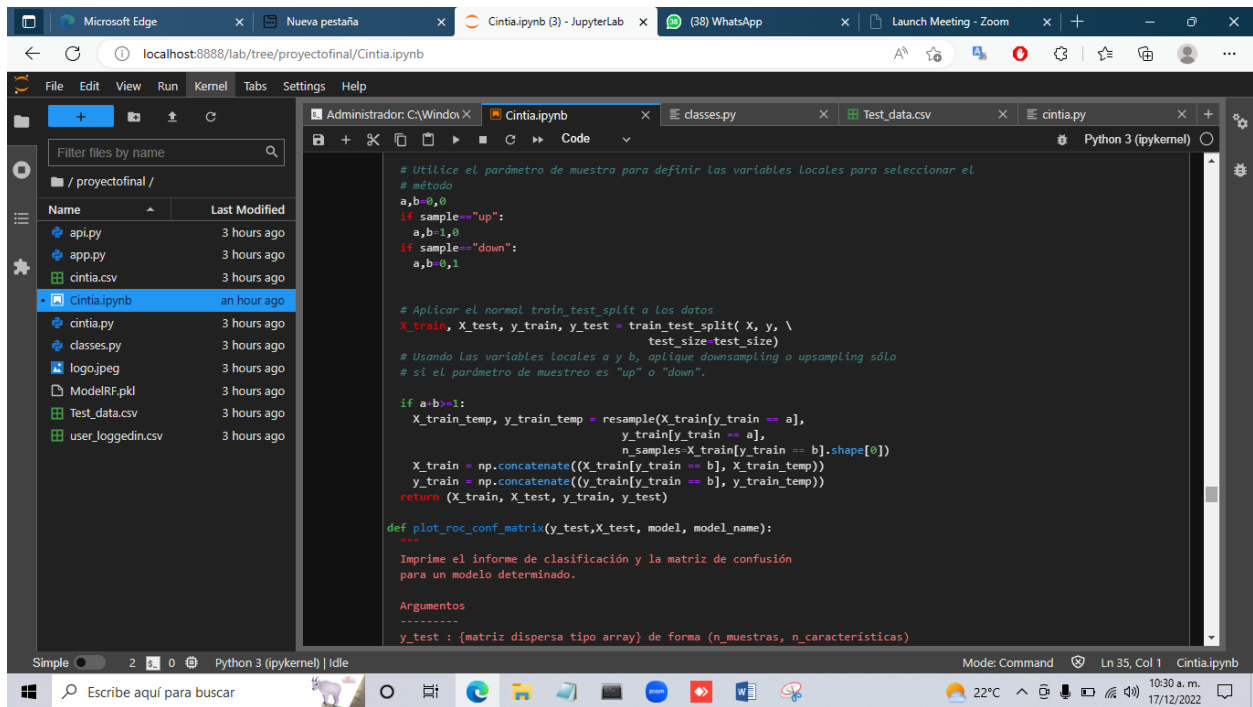


Fig 15. Entrenando Variables para el modelo

6. ESPECIFICACIONES DE REQUISITOS FUNCIONALES / NO FUNCIONALES

6.1. Requisitos Funcionales.

El dashboard debe poder mostrar graficas estadísticas aparte pueda realizar predicciones usando variables ya establecidas por el modelo de predicción, dicha predicción será el precio del vehículo con esas características de las variables.

Para su uso de manera local de los archivos, se debe instalar las librerías que se mencionan en los **requirements.txt** encontrados en sus respectivos repositorios de GitHub.

6.2. Requisitos No Funcionales.

- ❖ **Desarrollo:** Lenguaje de programación a usar, patrones de diseño y entorno de desarrollo.
- ❖ **Operaciones:** procedimientos operativos de cómo usar el software.
- ❖ **Éticos:** aseguran que el sistema sea fiable tanto para el usuario.

7. OBJETIVOS DEL SISTEMA.

7.1. Objetivo General.

Crear un aplicativo web utilizando las librerías de Python tales como Streamlit, Plotly Express, Pandas y Numpy que permita interactuar con un modelo de Machine Learning ya entrenado y posteriormente con los datos del dataset respectivamente asignado, siguiendo de manera correcta los mejores métodos para la comunicación y visualización efectiva de datos.

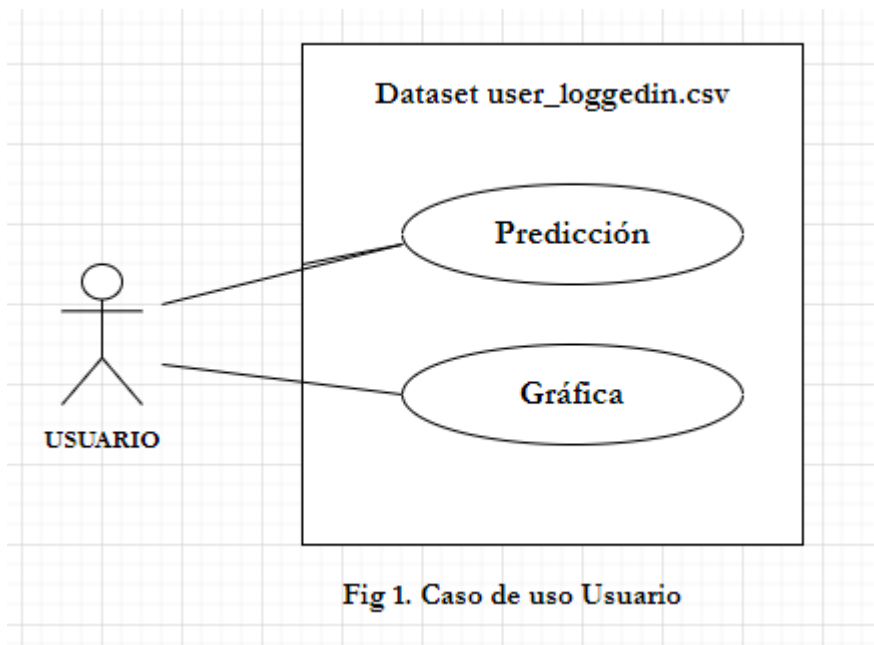
7.2. Objetivos Específicos.

- ❖ Implementar librerías y técnicas que permitan la representación de información de datos masivos utilizando un lenguaje de alto nivel que permita analizar datos de forma eficiente como lo es Python.
- ❖ Desarrollar un modelo de Machine Learning que permita predecir, dado unas variables, el precio de un vehículo usado.
- ❖ Establecer una conexión por medio de una API, el modelo con el dashboard.
- ❖ Diseñar con la librería Streamlit un dashboard funcional como parte de este proyecto.

8. UML

A continuación, se muestran los diagramas de Casos de Uso, Secuencia, actividades, componentes, de clases y Modelo Entidad-Relación en los que se basó el diseño y programación del sistema.

8.1. Casos de Uso:



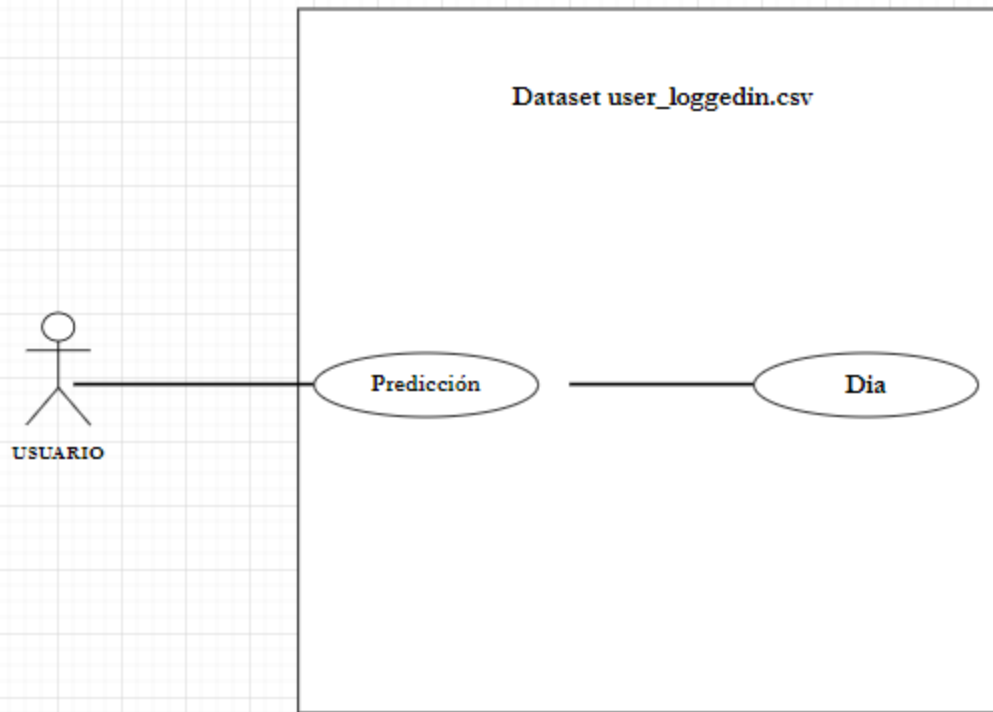


Fig 2. Caso de uso Diagrama de predicción

Usuario: El usuario al entrar al aplicativo puede mover y seleccionar los datos o variables que quiere que el modelo prediga que se encuentran en el Sidebar.

8.2. Diagrama de Actividades:

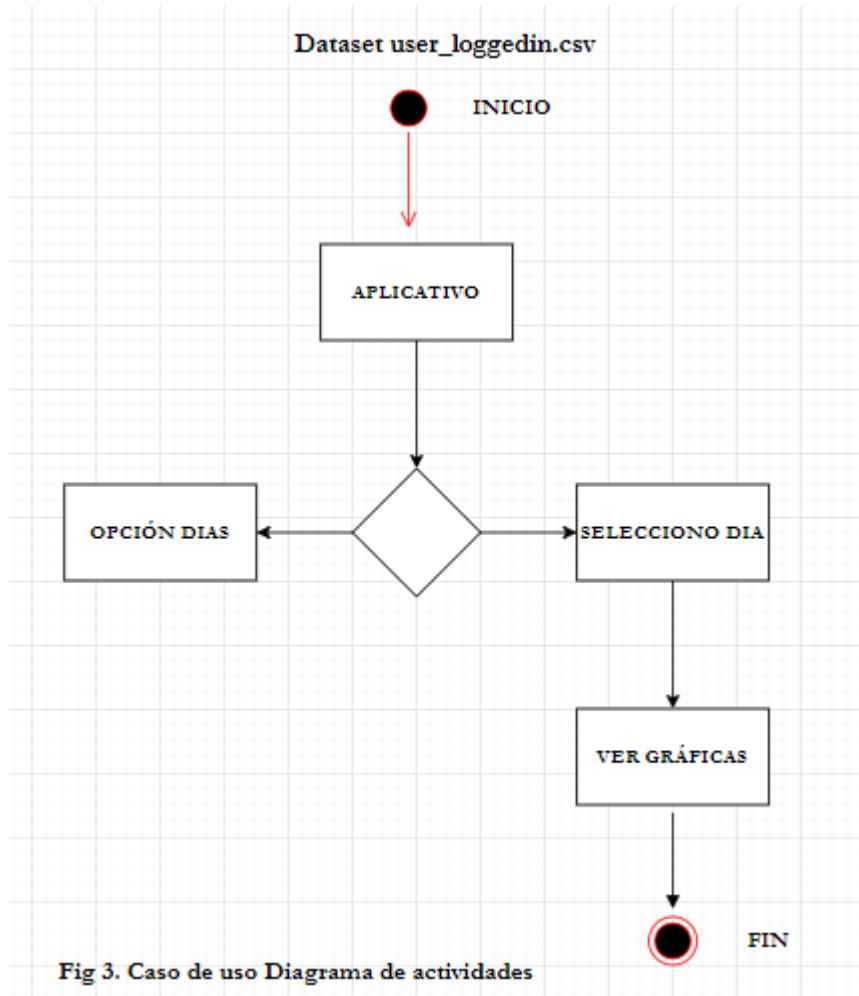


Fig 3. Caso de uso Diagrama de actividades

8.3. Diagrama Modelo Entidad-Relación:

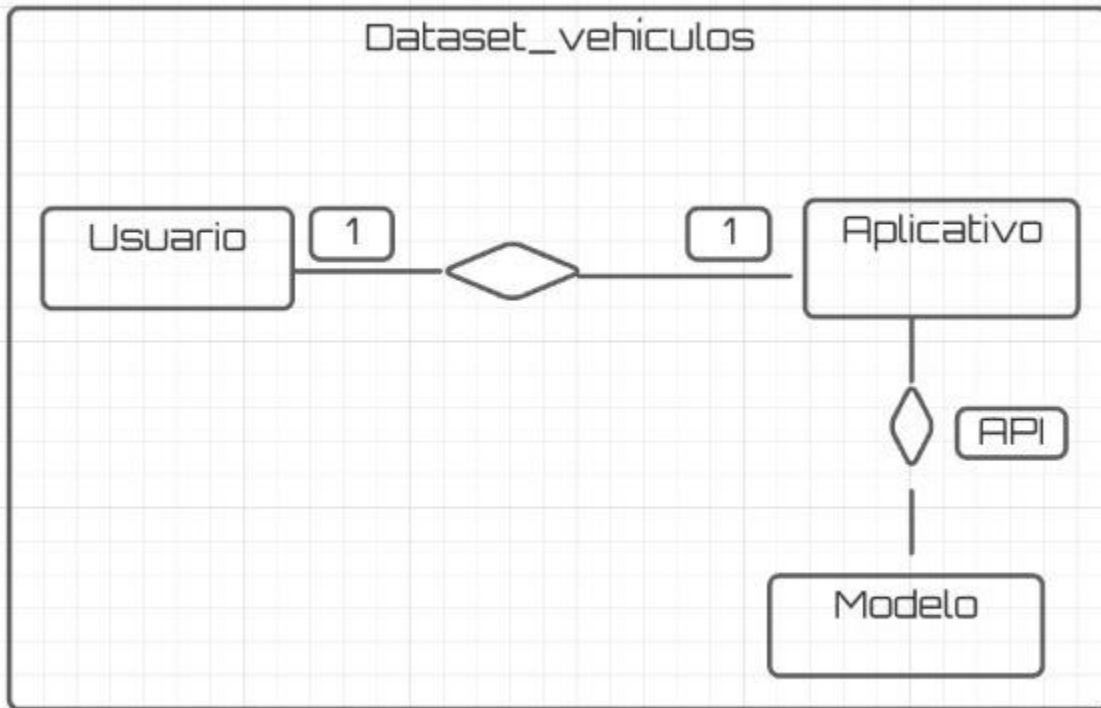
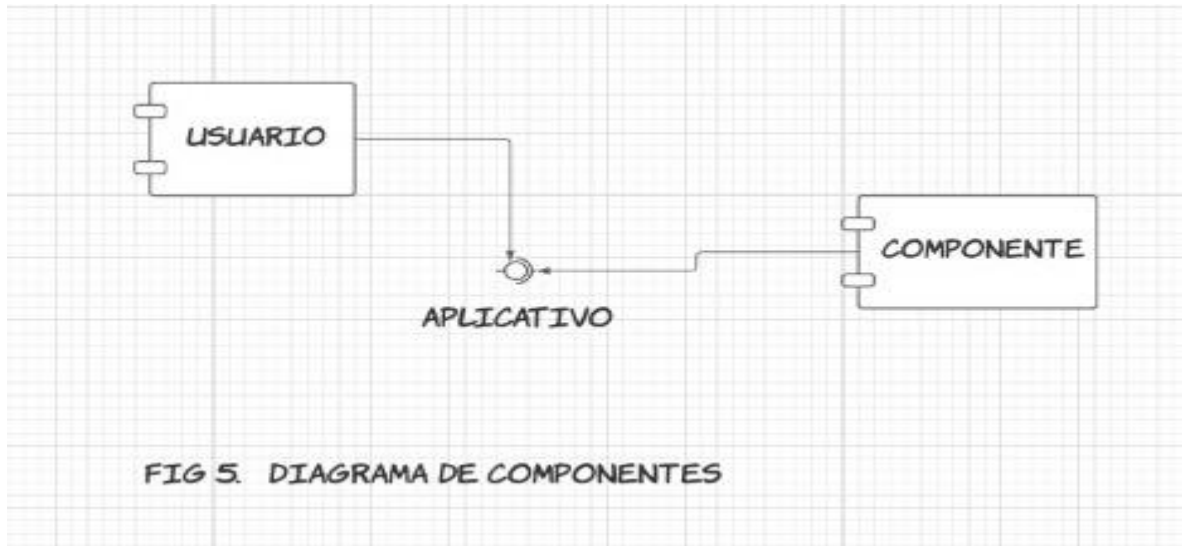


Fig 4. Modelo E/R

8.4. Diagrama de Componentes:



9. ANEXO.

9.1. MANUAL DE USUARIO.

1. INTRODUCCIÓN.

El presente documento se realizó con el fin de orientar a los usuarios sobre el aplicativo del proyecto final de nuestro diplomado, instruyéndolo en el uso adecuado y permitiendo que le sea realmente provechoso; por lo que la familiarización con el entorno se llevará a cabo de la mejor manera.

2. DESCRIPCIÓN.

En muchas ocasiones los usuarios de software no hacen un uso correcto de ellos, o más bien no utilizan todas las herramientas que se disponen allí para su trabajo; es por esto que una guía para instruirlos es fundamental.

En el Manual de usuario de este aplicativo, se presenta a los usuarios el debido la predicción y gráfico.

3. CONTENIDO.

A continuación, se presenta una introducción al sistema en donde se describirá su funcionamiento, permitiendo determinar la utilidad y usabilidad de este; luego en la sección de ingreso al sistema se presentarán los pasos determinados y adecuados que permitirán acceder a este.

En los componentes de interfaz, se evidencian las herramientas que pertenecen a este y la descripción del uso de cada una de estas; posteriormente se hará referencia al uso del sistema para que el usuario conozca cada parte del mismo, con sus respectivas pantallas.

4. INTRODUCCIÓN AL SISTEMA.

El aplicativo permite facilitar el uso de un modelo de predicción para ver cuantos estudiantes se loguean a una hora especificada

5. INGRESO AL SISTEMA.

Para acceder a la aplicación, el usuario debe de hacer uso del siguiente enlace:

<https://webdiplomado.herokuapp.com/>

Al acceder a la URL podrá ver la interfaz de entrada del aplicativo.



Fig 1. Observamos el menú de nuestro Dashboard

6. COMPONENTES DE INTERFAZ.

6.1. Inicio.

Una vez iniciado se haya accedido por medio del link aparecerá la vista de Inicio en la que se dispone:

1. **Menú (Sidebar):** Contiene las variables que cambian el valor del precio de la predicción.
2. **Gráficas:** Muestra los gráficos donde se muestran los distintos análisis basados en los datos almacenados en el sistema.



Fig 2. Observamos la cantidad de ID que se loguean en dicha hora en la plataforma Cintia

6.2. Menú (Sidebar):

1. **Selectores:** esta sección contiene todas las variables necesarias para realizar la predicción del precio de referencia, los cuales son identificación, hora, fecha y programa.
2. **Selector:** contiene un selector para cambiar las variables del grafico de barras.

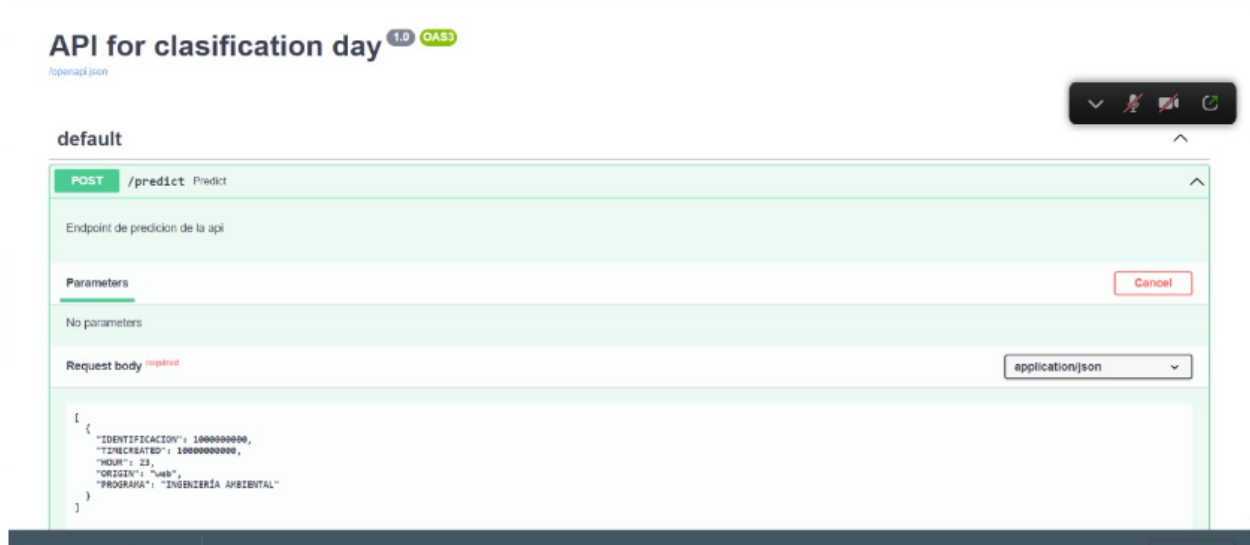


Fig 3. Observamos la api funcionando

7. USO DEL SISTEMA

7.1. Predicción.

Al momento Entrar al aplicativo en la vista de Inicio podemos ver el precio de la predicción.



La predicción del precio puede cambiar utilizando los selectores que se encuentran en el Sidebar. Al mismo momento que cambias los valores de los selectores el precio cambiará en base a esos valores. Se muestran ya precargados los datos que usan para la predicción.

7.2. Gráficas:

8. CONCLUSIÓN.

El anterior manual de usuario se espera que explique el funcionamiento del aplicativo presentado en este proyecto.

REFERENCIAS.

- Datamedia. (2012). *Datamedia*. Obtenido de Datamedia: <https://datademia.es/blog/ques-python>
- Martin, F. (03 de 12 de 2019). *martinFowler.com*. Obtenido de The New Methodology: <https://martinfowler.com/articles/newMethodology.html>
- Penadés, C., & Torres, P. O. (2006). Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). *Técnica administrativa*, 1666-1680.