

PROYECTO FINAL

Línea 1: Desarrollo de soluciones – Ágil

**Plataforma de Análisis Interactivo y Modelo de Predicción de Emisiones de Carbono en Taxis de NYC: Información Accesible para Decisiones Sostenibles y Basadas en Datos**

**Integrantes:**

Hospinal Roman, Oscar David (0000-0002-5298-6268)

Jofré, Leandro Gastón (0009-0002-3075-7883)

Oria García, Pedro Santos (0009-0004-8777-6991)

Alarcón Bothia, Ilbert Ferney (0009-0009-3556-3421)

Cáceres, Juliana (0009-0004-0482-109X)

**ASESOR**

Lila Alves

Jonathan Deiloff

**2023-02**

TABLA DE CONTENIDO

[TABLA DE CONTENIDO 2](#_Toc139243363)

[INDICE DE FIGURAS 4](#_Toc139243364)

[INDICE DE TABLAS 5](#_Toc139243365)

[CAPITULO 1: DEFINICIÓN DEL PROYECTO 8](#_Toc139243366)

[1.1 Definición del problema 8](#_Toc139243367)

[1.1.1 Objeto de estudio 8](#_Toc139243368)

[1.1.2 Dominio del Problema 8](#_Toc139243369)

[1.1.3 Lienzo Lean Canvas 9](#_Toc139243370)

[1.2 Definición de la solución 10](#_Toc139243371)

[1.2.1 Objetivos del proyecto 10](#_Toc139243372)

[1.2.2 Indicadores de éxito 11](#_Toc139243373)

[1.2.3 Alcance 12](#_Toc139243374)

[1.2.4 Restricción Funcional del alcance 13](#_Toc139243375)

[CAPITULO 3: DESARROLLO DE LA SOLUCIÓN 13](#_Toc139243376)

[3.1 Incepción Ágil 13](#_Toc139243377)

[3.1.1. Visión del Producto 13](#_Toc139243378)

[3.1.2 Definición de Roles 14](#_Toc139243379)

[3.1.3 Historias de Usuario 15](#_Toc139243380)

[3.1.4 Tablero Scrum Base. 19](#_Toc139243381)

[3.1.5 Diagrama de Gannt. 20](#_Toc139243382)

[3.1.6 Cronología del Proyecto. 21](#_Toc139243383)

[3.2 Análisis preliminar de calidad de datos 21](#_Toc139243384)

[3.3 Arquitectura de software 22](#_Toc139243385)

[3.3.1 Arquitectura Técnica Base 22](#_Toc139243386)

[3.4 Desarrollo de Sprint 1 24](#_Toc139243387)

[3.4.1 Sprint Planning 1 24](#_Toc139243388)

[3.4.2 Construcción del Sprint 1 26](#_Toc139243389)

[3.4.3 Daily Meeting 29](#_Toc139243390)

[3.4.4 Sprint Retrospective 31](#_Toc139243391)

[3.5 Desarrollo de Sprint 2 31](#_Toc139243392)

[3.5.1 Sprint Planning 2 31](#_Toc139243393)

[3.5.2 Construcción del Sprint 2 34](#_Toc139243394)

[3.5.3 Daily Meeting 40](#_Toc139243395)

[3.5.4 Sprint Retrospective 43](#_Toc139243396)

[CAPITULO 7: CONCLUSIONES DEL PROYECTO 43](#_Toc139243397)

[7.1 Conclusiones 43](#_Toc139243398)

[7.2 Recomendaciones 44](#_Toc139243399)

[GLOSARIO 45](#_Toc139243400)

[SIGLARIO 45](#_Toc139243401)

[REFERENCIAS BIBLIOGRÁFICAS 47](#_Toc139243402)

[ANEXOS 47](#_Toc139243403)

[I. Actas 47](#_Toc139243404)

[II. Gestión del Proyecto 48](#_Toc139243405)

INDICE DE FIGURAS

[**Figura 1.** Lienzo Canvas 10](#_Toc139706722)

[**Figura 2.** Visión del producto 14](#_Toc139706723)

[**Figura 3.** Historias de usuario 16](#_Toc139706724)

[**Figura 4.** Desglose del product Backlog 18](#_Toc139706725)

[**Figura 5.** Historia de usuario por épicas 19](#_Toc139706726)

[**Figura 6.** Tablero scrum base 20](#_Toc139706727)

[**Figura 7.** Diagrama de gannt 21](#_Toc139706728)

[**Figura 8.** Cronología del proyecto 22](#_Toc139706729)

[**Figura 9.** Arquitectura técnica Base 24](#_Toc139706730)

[**Figura 10.** Historia de usuario HU001-Epica01-Sprint01 26](#_Toc139706731)

[**Figura 11.** Mockup de la HU001 28](#_Toc139706732)

[28](#_Toc139706733)

[**Figura 12.** Product Backlog Refinado - Sprint01 29](#_Toc139706734)

[**Figura 13.** Burdown Chart -Sprint01 30](#_Toc139706735)

[**Figura 14.** Sprint Retrospective 01 32](#_Toc139706736)

[**Figura 15.** Historia de usuario HU002 -Epica01-Sprint02 33](#_Toc139706737)

[33](#_Toc139706738)

[**Figura 16.** Historia de usuario HU003 -Epica02-Sprint02 33](#_Toc139706739)

[34](#_Toc139706740)

[**Figura 17.** Mockup de la HU002 35](#_Toc139706741)

[**Figura 18.** Mockup de la HU003 35](#_Toc139706742)

[**Figura 19.** Product backlog -Sprint02 37](#_Toc139706743)

[**Figura 20.** Incremento del Sprint-HU002 38](#_Toc139706744)

[39](#_Toc139706745)

[**Figura 21.** Interfaz Mongo Compas 43](#_Toc139706746)

[**Figura 22.** Servicio de Atlas en MongoDB 44](#_Toc139706747)

[**Figura 23.** Estructura del Data warehouse 45](#_Toc139706748)

[**Figura 24.** Creación de Usuarios 45](#_Toc139706749)

[**Figura 25.** Dataset ETL\_AirQ.csv 49](#_Toc139706750)

[**Figura 26.** Dataset ETL\_C\_Combust.csv 49](#_Toc139706751)

[**Figura 27.** Dataset Taxis Rutas 50](#_Toc139706752)

[**Figura 28.** Dataset Taxis Zonas 50](#_Toc139706753)

[**Figura 29.** Dataset Sonora 51](#_Toc139706754)

[**Figura 30.** Workflow Detallado 52](#_Toc139706755)

[**Figura 31.** Gráfico de viajes por mes 53](#_Toc139706756)

[**Figura 32.** Gráfico de correlación sobre las zonas de New York. 53](#_Toc139706757)

[**Figura 33.** Gráfico de dispersión de la tabla de Combustión y CO2. 54](#_Toc139706758)

[**Figura 34.** Gráfico de barras sobre las emisiones por vehículo en New york. 54](#_Toc139706759)

[**Figura 35.** Gráfico de Línea de los ingresos por mes. 55](#_Toc139706760)

[**Figura 36.** Sprint Burdonwn Chart -Sprint 02 57](#_Toc139706761)

[**Figura 37.** Sprint Retrospective 02 58](#_Toc139706762)

INDICE DE TABLAS

[**Tabla 1.** Indicadores de éxito. 14](#_Toc139706306)

[**Tabla 2.** Análisis de calidad de datos 25](#_Toc139706307)

[**Tabla 3.** Resumen sprint planning 1 29](#_Toc139706308)

[**Tabla 4.** Sprint 01 – Revisión del progreso. 32](#_Toc139706309)

[**Tabla 5.** Resumen sprint planning 2 36](#_Toc139706310)

[**Tabla 6.** Diseño del modelo conceptual de datos 41](#_Toc139706311)

[**Tabla 7.** Detalle de los Pipelines 43](#_Toc139706312)

[**Tabla 8.** Tabla detalle de las Reglas de Negocio 48](#_Toc139706313)

[**Tabla 9.** Tabla detalle de la Validación de scripts de la regla del Negocio 49](#_Toc139706314)

[**Tabla 10.** Sprint 02 – Revisión del progreso. 58](#_Toc139706315)

[**Tabla 11.** Glosario de términos usados en el documento. 62](#_Toc139706316)

[**Tabla 12.** Siglario de las siglas usadas en el documento. 63](#_Toc139706317)

RESUMEN

El proyecto "NYC Taxis & Carbon Emission" es una iniciativa de análisis de datos con enfoque en el sector de transporte de taxis de la ciudad de Nueva York y su relación con la contaminación del aire y la sonora. La idea central es entender los patrones de uso de los taxis, las emisiones que generan y cómo estos factores influyen en la calidad del aire y la contaminación sonora de la ciudad.

La empresa de servicios de transporte de pasajeros contratante está interesada en expandir su negocio hacia el sector de taxis y busca entender el panorama actual para tomar decisiones informadas. Su visión de futuro contempla la posibilidad de introducir vehículos eléctricos en su flota, lo que podría disminuir la contaminación y mejorar la calidad del aire.

Para lograr los objetivos del proyecto, se utilizarán diversas fuentes de datos, incluyendo datos de viajes de taxis, condiciones climáticas, contaminación sonora, contaminación del aire, emisiones de CO2 y otros factores relevantes. La estrategia de análisis implica la recopilación y limpieza de estos datos, el análisis estadístico para identificar tendencias y patrones, y la implementación de modelos de aprendizaje automático para realizar predicciones y proporcionar recomendaciones basadas en los hallazgos.

Finalmente, las métricas y los indicadores clave de rendimiento (KPI) se utilizarán para evaluar la eficacia del proyecto y guiar futuras decisiones. El objetivo final es proporcionar un conjunto de datos y análisis robustos que la empresa de transporte pueda utilizar para mejorar sus servicios y contribuir a un futuro menos contaminado.

# CAPITULO 1: DEFINICIÓN DEL PROYECTO

## Definición del problema

El problema central es la falta de información sobre las emisiones de los taxis y su impacto ambiental, lo que impide a la empresa de transporte tomar decisiones estratégicas para reducir la contaminación y mejorar la calidad del aire en la ciudad de Nueva York. Esta falta de información dificulta la implementación de políticas y medidas efectivas para abordar el problema de la contaminación del aire causada por los taxis.

Además, sin información detallada sobre las emisiones de los taxis, es difícil establecer metas realistas y medibles para reducir la contaminación y monitorear el progreso en el tiempo. La empresa podría implementar medidas generales, como reemplazar gradualmente su flota por vehículos eléctricos, pero sin una comprensión precisa de las emisiones actuales y las áreas donde se concentran los niveles más altos de contaminación, es difícil optimizar los esfuerzos y asignar recursos de manera eficiente.

### Objeto de estudio

El objetivo principal es analizar los datos relacionados con los viajes de taxis, las condiciones climáticas, la contaminación sonora, la contaminación del aire, las emisiones de CO2 y otros factores relevantes. Se busca recopilar y limpiar estos datos para realizar un análisis estadístico y aplicar modelos de aprendizaje automático con el fin de identificar tendencias, patrones y realizar predicciones.

### Dominio del Problema

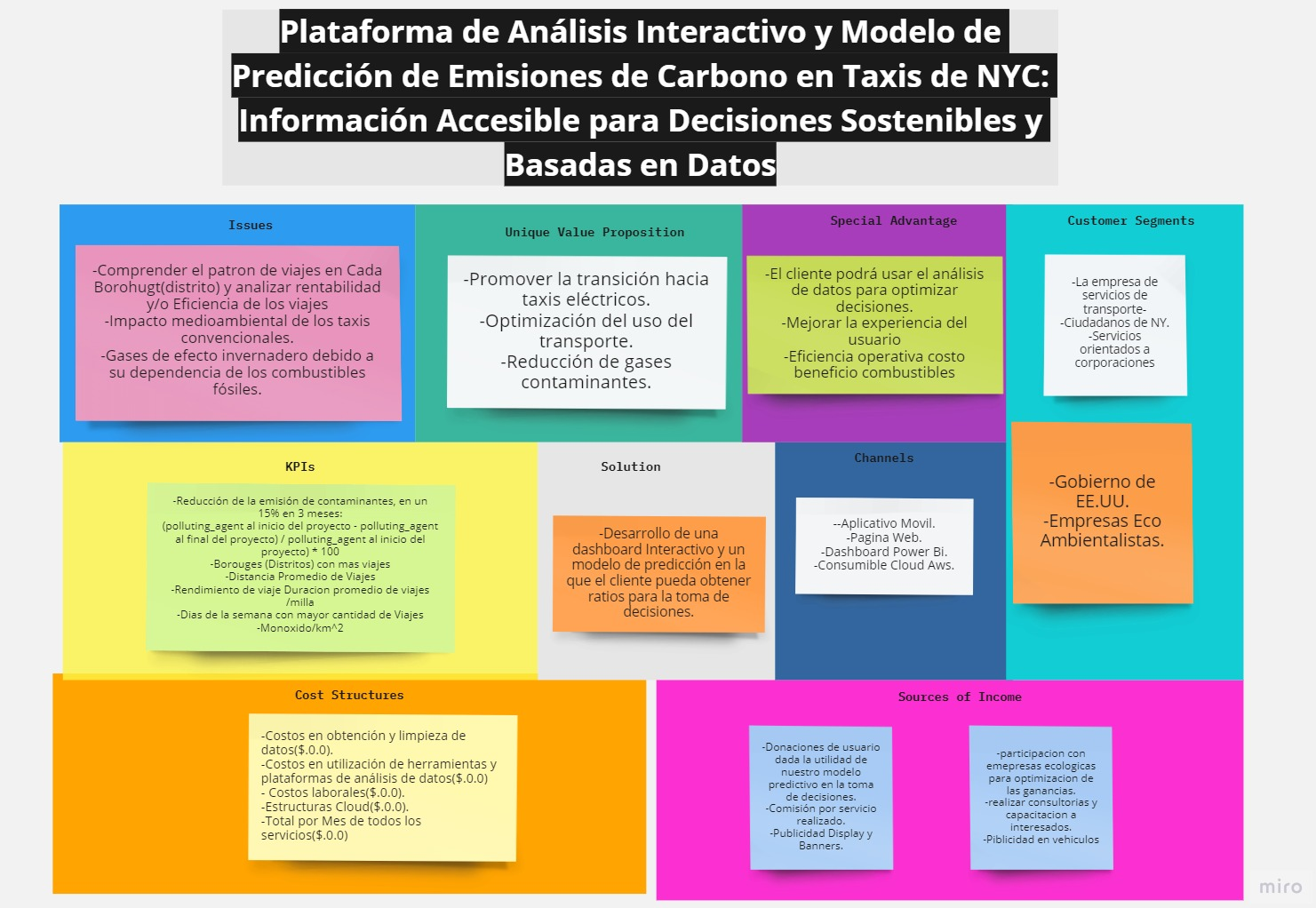
Si bien la contaminación del aire en general es un problema importante en las áreas urbanas, el enfoque principal de este proyecto se centra en las emisiones de CO2 debido a su papel significativo en el calentamiento global y el cambio climático. El CO2 es uno de los principales gases de efecto invernadero que contribuyen al atrapamiento del calor en la atmósfera, lo que conduce al calentamiento global y sus consecuencias asociadas.

En este proyecto en particular, nos enfocaremos específicamente en la contaminación de dióxido de carbono (CO2) en la ciudad de Nueva York. Si bien el dominio del problema abarca la contaminación del aire y la contaminación sonora en el transporte de taxis.

### Lienzo Lean Canvas

Se trabajó con un lienzo Lean Canvas en el proyecto porque nos proporciona una estructura clara y concisa para definir la propuesta de valor, identificar oportunidades y desafíos, y mantener un enfoque ágil y centrado en el cliente. Ayuda a comunicar de manera efectiva la visión del proyecto y a tomar decisiones informadas para lograr el éxito.

1. Lienzo Canvas



Nota. El presente diagrama representa la visión final del proyecto

## Definición de la solución

El proyecto consiste en desarrollar una solución integral que incluye un dashboard interactivo y un modelo de predicción. El objetivo principal es brindar al cliente la capacidad de obtener ratios y métricas relevantes para la toma de decisiones relacionadas con el sector de taxis en la ciudad de Nueva York.

La solución permitirá visualizar de manera intuitiva y en tiempo real diversos indicadores clave, como la duración de viajes, porcentaje de tarifas, viajes inter e intra boroughs, días y semanas con mayor cantidad de viajes, distancia promedio por pasajero, propinas promedio por pasajero, tipos de pago más utilizados, entre otros.

Además, se desarrollará un modelo de predicción basado en técnicas de machine learning, que utilizará datos históricos y variables relevantes para estimar las emisiones de carbono generadas por los taxis. Esto permitirá al cliente tener una visión clara del impacto ambiental de su flota de vehículos y evaluar la viabilidad de implementar vehículos eléctricos o tomar medidas para reducir las emisiones.

La solución se basará en una plataforma en la nube para garantizar el acceso remoto y la escalabilidad, y se utilizarán tecnologías como Hugging Face para el desarrollo del modelo de predicción y Power BI para la creación del dashboard interactivo.

Con esta solución, el cliente podrá obtener información en tiempo real, identificar patrones y tendencias, y tomar decisiones informadas y sostenibles para mejorar la eficiencia y calidad del servicio de taxis, al tiempo que se consideran aspectos ambientales y de sostenibilidad.

### Objetivos del proyecto

#### Objetivo General

Es de optimizar los servicios de taxis en NYC, identificando patrones de demanda y oportunidades de mejora. Se desarrollará un modelo de pronóstico con una precisión del 98% para estimar las emisiones de carbono, impulsando decisiones más sostenibles y eficientes.

#### Objetivos específicos

Por cada sprint se cumplirá lo siguiente:

* **OE-01:** Definir el alcance del proyecto y establecer la infraestructura necesaria, utilizando metodologías ágiles. Colaborar en la organización y distribución de tareas, asignando roles específicos y creando el repositorio de código, fuentes de datos y documentación.
* **OE-02:** Implementar pipelines de extracción, transformación y carga (ETL) de datos hacia estructuras de almacenamiento, como Data Warehouse, Data Lake o Data Lakehouse. Considerar la carga incremental de datos y utilizar herramientas de big data y/o servicios en la nube para optimizar el proceso.
* **OE-03:** Desarrollar un dashboard interactivo y un modelo de Machine Learning utilizando los datos procesados. Incluir los KPIs relevantes para el análisis realizado y preparar un storytelling efectivo para presentar los resultados. Elaborar reportes y dashboards, definir y calcular los KPIs, y poner en producción el modelo de Machine Learning.

### Indicadores de éxito

El cumplimiento de los objetivos del proyecto se mide a través de los indicadores de logro detallados en la siguiente tabla:

1. Indicadores de éxito.

|  |  |  |
| --- | --- | --- |
| **Indicador de éxito** | | **Objetivo** |
| I-01 | **Reducción de la emisión de contaminantes.**  Objetivo: Reducir la emisión de contaminantes en un 15% en 3 meses.  **Formula**  **%PR= (Pi –Pf)/(Pi \*100)**  *Donde:*  *PR: Reducción de la emisión de contaminantes.*  *Pi: Polución al inicio (polluting\_agent al inicio del período).*  *Pf: Polución al final (polluting\_agent al final del período).* | **OE-01** |
| I-02 | **Aumento de viajes compartidos**  Objetivo: Aumentar los viajes compartidos en un 10% en 3 meses.  **Formula**  **%AT = ((Tf - Ti) / Ti) \* 100**  Donde:  AT: Aumento de viajes compartidos.  Ti: Porcentaje de viajes compartidos al inicio (trips\_per\_day\_shared al inicio del período).  Tf: Porcentaje de viajes compartidos al final (trips\_per\_day\_shared al final del período). | **OE-02** |
| I-03 | **Mejora de la eficiencia de los conductores**  Objetivo: Aumentar las horas promedio de trabajo por conductor en un 5% en 3 meses.  **Formula**  **%ED = ((Df - Di) / Di) \* 100**  *Donde:*  *ED: Eficiencia del conductor.*  *Di: Horas promedio de trabajo por conductor al inicio (avg\_hours\_per\_day\_per\_driver al inicio del período).*  *Df: Horas promedio de trabajo por conductor al final (avg\_hours\_per\_day\_per\_driver al final del período).* | **OE-03** |
| I-04 | **Aumento de los ingresos**  Objetivo: Aumentar el farebox\_per\_day en un 10% en 3 meses.  **Formula**.  **%RI = ((If - Ii) / Ii) \* 100**  *Donde:*  *RI: Aumento de los ingresos.*  *Ii: Ingresos al inicio (farebox\_per\_day al inicio del período).*  *If: Ingresos al final (farebox\_per\_day al final del período).* | **OE-03** |

### Alcance

El proyecto de taxis incluye la recolección y procesamiento de datos, el diseño e implementación de un Data Warehouse, el análisis exploratorio de datos, el desarrollo de modelos de Machine Learning, la creación de un dashboard interactivo y la implementación del modelo de Machine Learning en producción. Estas tareas se llevarán a cabo siguiendo una metodología ágil, con roles y responsabilidades asignados, reuniones periódicas de seguimiento (Daily) y herramientas colaborativas para facilitar la comunicación y el intercambio de información. El proyecto se desarrollará dentro de un marco de tiempo definido de 3 semanas, con hitos y entregables establecidos en la documentación de HENRY, y se gestionarán los riesgos de manera proactiva para maximizar el valor entregado a los Stakeholders.

### Restricción Funcional del alcance

Las restricciones de este proyecto incluyen un plazo de tiempo específico de 3 semanas para su finalización, el desarrollo de este proyecto se limita a la ciudad de New York no se puede ampliar su aplicación a otras ciudades, este proyecto no contempla factores topográficos o de tránsito. Los recursos de cómputo son limitados conforme al stack tecnologico, asi como las consideraciones de los stakeholders identificados inicialmente, adicional el proyecto cumple con las leyes y regulaciones, y la posibilidad de ajustar el alcance ante cambios significativos en el negocio. Estas restricciones se establecen para garantizar una gestión efectiva y exitosa del proyecto, optimizando los recursos disponibles y asegurando el cumplimiento de los objetivos dentro de los límites establecidos.

# CAPITULO 3: DESARROLLO DE LA SOLUCIÓN

## Incepción Ágil

### Visión del Producto

En este punto, se obtiene la estrategia general del negocio con el fin de establecer las necesidades, los usuarios y los beneficios del producto al negocio.

1. Visión del producto

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Nota. El presente diagrama es la estrategia general del negocio

### Definición de Roles

1. **Data Engineers:**

* Leandro Gastón Jofré: Responsable de la recopilación, limpieza y transformación de los datos necesarios para el proyecto "NYC Taxis & Carbon Emission". Trabaja en la implementación y mantenimiento de los sistemas de almacenamiento y procesamiento de datos.
* Oria García Pedro Santos: Encargado de desarrollar y mantener las infraestructuras de datos necesarias para el proyecto. Colabora en la implementación de flujos de datos eficientes y en la integración de diversas fuentes de datos.

1. **Data Scientist:**

* Ilbert Ferney Alarcón Bothia: Especialista en análisis de datos y modelos de aprendizaje automático. Se encarga de aplicar técnicas estadísticas y algoritmos de machine learning para descubrir patrones, identificar tendencias y realizar predicciones relacionadas con los patrones de uso de los taxis y su impacto en la contaminación del aire y la contaminación sonora.

1. **Data Analyst:**

* Juliana Caceres: Responsable de analizar los datos recopilados y realizar análisis exploratorios para extraer información relevante y generar insights que ayuden en la toma de decisiones. Trabaja en la identificación de métricas clave y en la generación de informes y visualizaciones para comunicar los hallazgos de manera efectiva.

1. **Project Manager:**

* Hospinal Roman Oscar David: Encargado de liderar y coordinar el proyecto "NYC Taxis & Carbon Emission". Gestiona el alcance, los recursos y los plazos del proyecto, además de facilitar la comunicación entre los diferentes roles. Supervisa la ejecución del proyecto y asegura que se cumplan los objetivos establecidos.

Cada uno de estos roles desempeña funciones específicas y contribuye de manera integral al éxito del proyecto, desde la recopilación y procesamiento de datos, pasando por el análisis y modelado, hasta la gestión general del proyecto. Trabajando en equipo, estos profesionales colaboran para generar un conjunto de datos y análisis robustos que permitan mejorar los servicios de transporte y contribuir a un futuro menos contaminado.

### Historias de Usuario

Como descripción de las historias de usuario, en donde se especifica el rol, la necesidad funcional y el valor o beneficio para el negocio.

1. Historias de usuario

Una captura de pantalla de una computadora

Descripción generada automáticamente

#### Product Backlog.

El producto backlog del proyecto, enumera todas las características, funcionalidades, requisitos, mejoras y correcciones que constituyen cambios a realizarse sobre el producto para entregas futuras. Los elementos de la “Lista de Producto” tienen como atributos la descripción, el orden, la estimación y el valor. Es así como, según las técnicas y metodologías explicadas anteriormente, es que se ha llegado a refinar nuestra lista de productos hasta conseguir una proporcionalidad según la prioridad requerida. A continuación, se detalla el refinamiento del Product Backlog asociado.

1. Desglose del product Backlog



1. Historia de usuario por épicas

Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

### Tablero Scrum Base.

Para la realización del tablero scrum base, se está utilizando la herramienta “Click Up”, en la que se consideran los estados: por hacer (to do), haciendo (doing) y terminado (done) como se detalla. Se debe tener en cuenta que, para realizar la definición del tablero, se hizo uso de las historias de usuario ya establecidas en el producto backlog.

1. Tablero scrum base

Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente

### Diagrama de Gannt.

La razón de utilizar un diagrama de Gantt en el proyecto es para visualizar y planificar de manera efectiva las tareas, el tiempo y la secuencia de actividades a lo largo de todo el proyecto. El diagrama de Gantt permite mostrar las fechas de inicio y finalización de cada tarea, así como las dependencias entre ellas, lo que facilita la coordinación y el seguimiento del progreso del proyecto. Además, el diagrama de Gantt proporciona una visión clara del cronograma del proyecto, permitiendo identificar posibles superposiciones de tareas, retrasos o adelantos, y ajustar el plan en consecuencia. En resumen, el diagrama de Gantt es una herramienta visual esencial para la planificación y gestión eficiente del proyecto, asegurando que se cumplan los plazos y se logren los objetivos establecidos.

1. Diagrama de gannt

Aplicación

Descripción generada automáticamente con confianza media

### Cronología del Proyecto.

La cronología del proyecto se desarrollará en distintas etapas, siguiendo un enfoque ágil y flexible. A continuación, se presenta un resumen de la cronología:

1. Cronología del proyecto

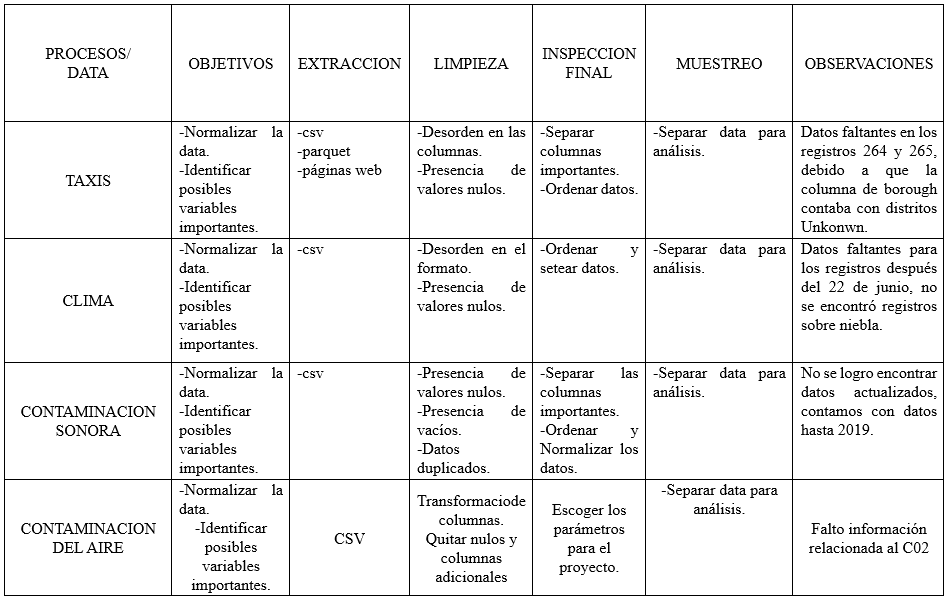
Interfaz de usuario gráfica, Aplicación, Tabla

Descripción generada automáticamente

## Análisis preliminar de calidad de datos

El análisis de calidad de datos en el proyecto "Plataforma de Análisis Interactivo y Modelo de Predicción de Emisiones de Carbono en Taxis de NYC" es un proceso fundamental que garantiza la precisión, consistencia e integridad de los datos antes de que sean utilizados para el modelado y la toma de decisiones.

1. Análisis de calidad de datos



## Arquitectura de software

El resultado de los esfuerzos relacionados con el diseño de arquitectura de software de alto nivel se desarrollará en este capítulo. Se detallarán los objetivos y las limitaciones, los mecanismos arquitectónicos, las perspectivas lógicas, la implementación y la entrega, así como la prueba de concepto para esta arquitectura de software.

### Arquitectura Técnica Base

#### 3.2.1.2 Arquitectura Técnica Base TO-BE

El punto "Arquitectura Técnica Base TO-BE" se refiere a la descripción de la arquitectura técnica que se utilizará en el proyecto. A continuación, se detallan las tecnologías que se utilizarán en cada etapa:

**1.ETL (Extracción, Transformación y Carga de datos):**

. Databricks: Plataforma que proporciona un entorno de trabajo colaborativo para realizar tareas de ETL y análisis de datos utilizando herramientas como PySpark.

. Jupyter Notebook: Entorno interactivo de programación que permite escribir y ejecutar código en Python para realizar tareas de extracción y transformación de datos.

. Beautiful Soup: Biblioteca de Python utilizada para el scraping web, permitiendo extraer datos de páginas HTML o XML.

Visual Studio Code: Entorno de desarrollo integrado (IDE) utilizado para escribir y editar el código en Python.

**2.Deploy y despliegue:**

.Base de datos en formato .parquet: Se utilizará el formato de archivo Parquet para almacenar los datos procesados, ya que ofrece una alta compresión y un acceso eficiente a los datos.

. Conexión a un S3 de AWS: Se establecerá una conexión con el servicio de almacenamiento en la nube Amazon S3 para almacenar los archivos Parquet y facilitar el acceso y la escalabilidad de los datos.

. Modelo en Machine Learning: Se utilizará un modelo de Machine Learning desarrollado previamente para estimar las emisiones de carbono en los taxis de NYC.

. Plataforma de Hugging Face: Se aprovechará la plataforma de Hugging Face para implementar y gestionar el modelo de Machine Learning, brindando una interfaz para su uso y despliegue eficiente.

**3.Output: Dashboard en Power BI**

Power BI: Herramienta de visualización de datos que permitirá crear un dashboard interactivo y dinámico para presentar los resultados del análisis de datos y las predicciones del modelo de Machine Learning.

En resumen, la arquitectura técnica propuesta utiliza tecnologías como Databricks, Jupyter Notebook, Beautiful Soup, Visual Studio Code, Parquet, AWS S3, Hugging Face y Power BI para realizar el ETL de los datos, almacenarlos en un formato eficiente, desplegar el modelo de Machine Learning y finalmente presentar los resultados a través de un dashboard interactivo en Power BI.

1. Arquitectura técnica Base

Interfaz de usuario gráfica, Diagrama

Descripción generada automáticamente

## Desarrollo de Sprint 1

### 3.4.1 Sprint Planning 1

#### Definition of Done (DoD)

A fin de garantizar la calidad del incremento y de cumplir con las condiciones requeridas por el producto, se debe cumplir lo siguiente:

* Todas las pruebas unitarias y funcionales deben ser correctas y validadas por los desarrolladores.
* El código debe estar completo de acuerdo con los estándares de desarrollo del equipo.
* El código de desarrollo del aplicativo debe estar versionado y en GitHub.
* El despliegue del proyecto debe estar en un entorno Dev.
* Todos los criterios de aceptación deben cumplirse.
* Todos los bugs deben estar corregidos.
* Los módulos dentro del alcance del sprint 1 deben ser aceptados por el Product Owner.

#### Historias de usuario para el Sprint 1

Las historias de usuario pertenecientes al Sprint 01, están compuesta por HU001 (Limpieza de datos), dentro de las cuales, se detallará por cada una los puntos de historia, criterios de aceptación y tareas técnicas correspondientes.

1. Historia de usuario HU001-Epica01-Sprint01

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

#### Resumen del Sprint Planning 1

La duración del Sprint 1 tiene una duración total de 1 semana.

1. Resumen sprint planning 1

|  |  |  |
| --- | --- | --- |
| ***Fecha Inicio del Sprint*** | ***Fecha Fin del Sprint*** | ***Días*** |
| 26/06/2023 | 30/06/2023 | 05 |

#### Sprint Backlog 1

Dentro del Sprint Backlog se detallan las tareas técnicas específicas para cada historia de usuario.

#### Tablero Scrum

El avance de la historia de usuario especificadas para el Sprint 01, se encuentra concluidas en fecha y presenta el detalle.

### 3.4.2 Construcción del Sprint 1

#### Diseño de prototipos

Los prototipos pertenecientes a cada historia de usuario, es decir, HU001, se detallan como prototipos realizados y con componentes actualizados.

Aquí se muestra un diagrama de flujo o diagrama de proceso que detalla las etapas de limpieza de datos y las decisiones que se toman en cada etapa. Por ejemplo, eliminación de duplicados, tratamiento de valores nulos, normalización, etc.

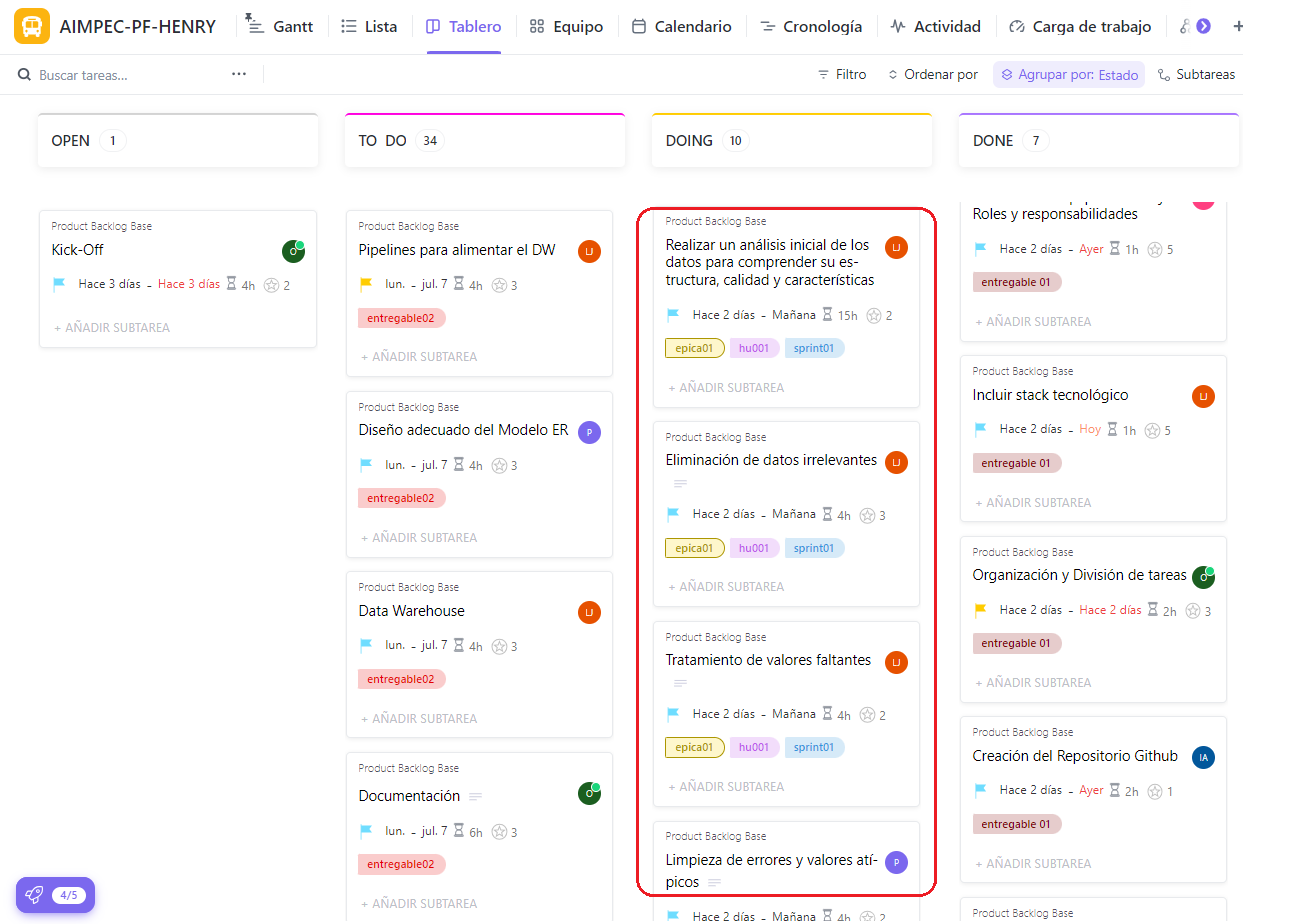
1. Mockup de la HU001

Diagrama

Descripción generada automáticamente

#### Product Backlog Refinado

El Product Backlog Refinado del Sprint 01 proporciona una visión clara de las tareas a realizar y establece una base sólida para el inicio del desarrollo durante el primer sprint.

1.   
   Product Backlog Refinado - Sprint01

### 3.4.3 Daily Meeting

#### Gestión del tablero en el Sprint

* La frecuencia de las reuniones será de cinco veces por semana, de lunes a viernes.
* Cada reunión tendrá una duración máxima de 45 minutos.
* Durante la reunión se realizará la revisión de los avances presentes y futuros por cada historia de usuario.

Revisión del progreso. Se han realizado todas las historias de usuario con un progreso concluido del 100% tal como se especifica:

1. Sprint 01 – Revisión del progreso.

|  |  |  |
| --- | --- | --- |
| **Historia de Usuario** | **Progreso** | **Estado** |
| HU001 | 100% | Terminado |

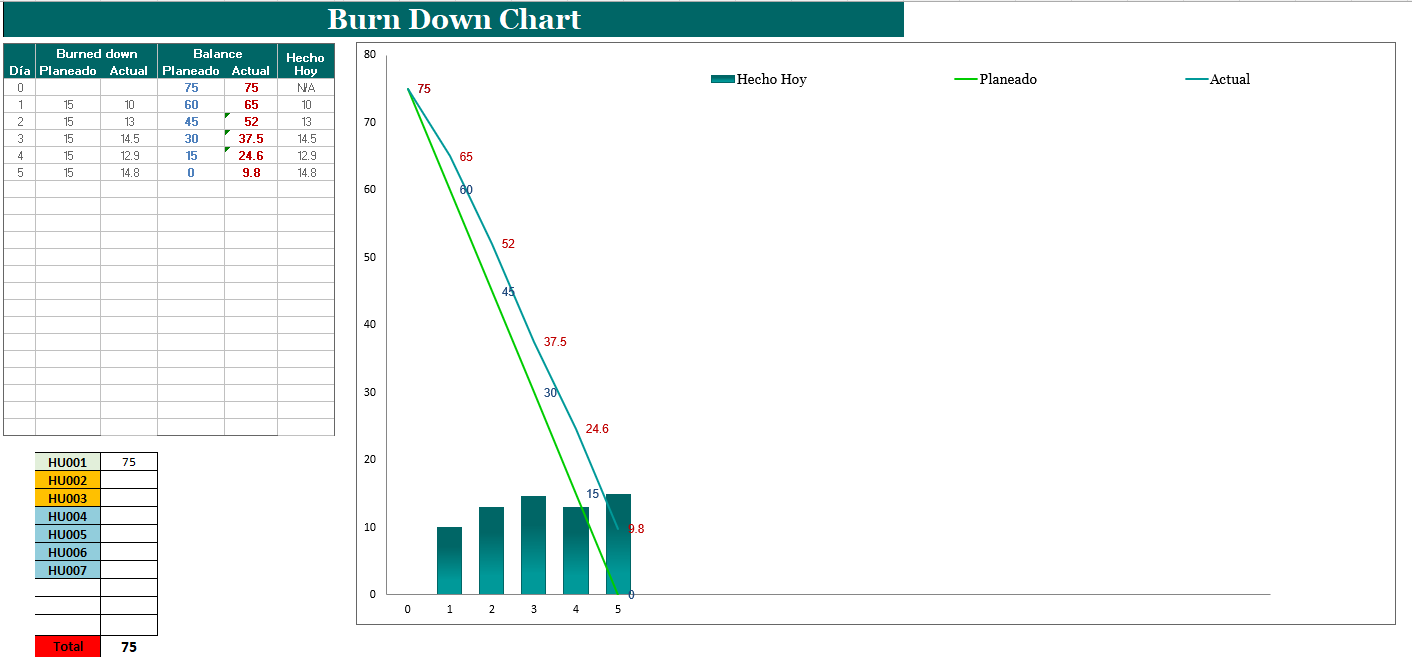
Identificación de impedimentos, bloqueos, dependencias y riesgos. Dentro de estas, se encuentran los siguientes impedimentos:

* Zonas horarias diferentes entre los desarrolladores del proyecto.

#### Sprint Burndown Chart

El gráfico de Burndown Chart se ingresa diariamente, después de cada dayli meeting y está realizado por el Scrum Master del proyecto con la finalidad de representar el esfuerzo esperado versus el real para el Sprint 1. Para poder obtener el porcentaje de avance diario, se toman los puntos de historia definidos en el Product Backlog, y se establece el esfuerzo diario obtenido.

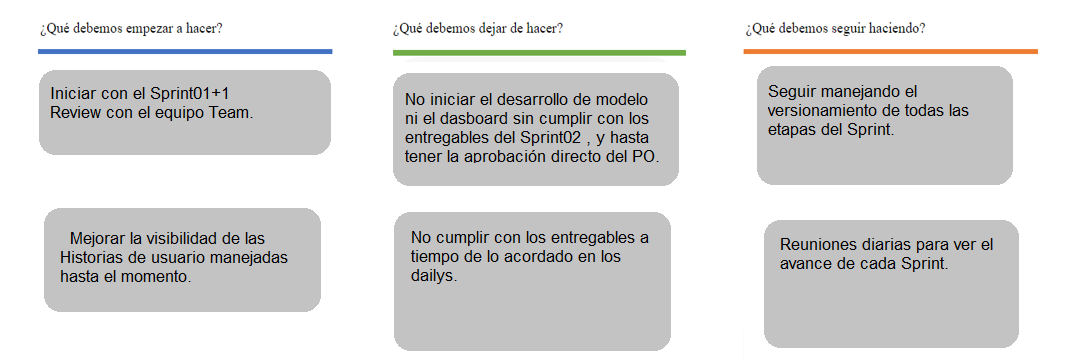
1. Burdown Chart -Sprint01



### 3.4.4 Sprint Retrospective

Dentro de los planes de mejora contemplados en la reunión realizada el día 30/06/2023, se obtuvieron los puntos detallados.

1. Sprint Retrospective 01



## Desarrollo de Sprint 2

### 3.5.1 Sprint Planning 2

#### Definition of Done (DoD)

A fin de garantizar la calidad del incremento y de cumplir con las condiciones requeridas por el producto, se debe cumplir lo siguiente:

* Todas las pruebas unitarias y funcionales deben ser correctas y validadas por los desarrolladores.
* El código debe estar completo de acuerdo con los estándares de desarrollo del equipo.
* El código de desarrollo del aplicativo debe estar versionado y en GitHub.
* El despliegue del proyecto debe estar en un entorno Dev.
* Todos los criterios de aceptación deben cumplirse.
* Todos los bugs deben estar corregidos.
* Los módulos dentro del alcance del sprint 2 deben ser aceptados por el Product Owner.

#### Historias de usuario para el Sprint 2

Las historias de usuario pertenecientes al Sprint 02, están compuesta por HU002 (Integración de datos) y HU003(Análisis exploratorio de datos), dentro de las cuales, se detallará por cada una los puntos de historia, criterios de aceptación y tareas técnicas correspondientes.

1. Historia de usuario HU002 -Epica01-Sprint02

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

1. Historia de usuario HU003 -Epica02-Sprint02

Interfaz de usuario gráfica, Texto, Aplicación, Teams

Descripción generada automáticamente

#### Resumen del Sprint Planning 2

La duración del Sprint 2 tiene una duración total de 1 semana.

1. Resumen sprint planning 2

|  |  |  |
| --- | --- | --- |
| ***Fecha Inicio del Sprint*** | ***Fecha Fin del Sprint*** | ***Días*** |
| 03/07/2023 | 07/07/2023 | 05 |

#### Sprint Backlog 2

Dentro del Sprint Backlog se detallan las tareas técnicas específicas para cada historia de usuario.

#### Tablero Scrum

El avance de la historia de usuario especificadas para el Sprint 02, se encuentra concluidas en fecha y presenta el detalle.

### 3.5.2 Construcción del Sprint 2

#### Diseño de prototipos

Los prototipos pertenecientes a cada historia de usuario, es decir, HU002 y HU003, se detallan como prototipos realizados y con componentes actualizados. Aquí se muestra un diagrama de flujo o diagrama de proceso que detalla las etapas de integración y análisis exploratorio de datos y las decisiones que se toman en cada etapa.

1. Diagrama

   Descripción generada automáticamente con confianza media  
   Mockup de la HU002
2. Mockup de la HU003

Imagen en blanco y negro

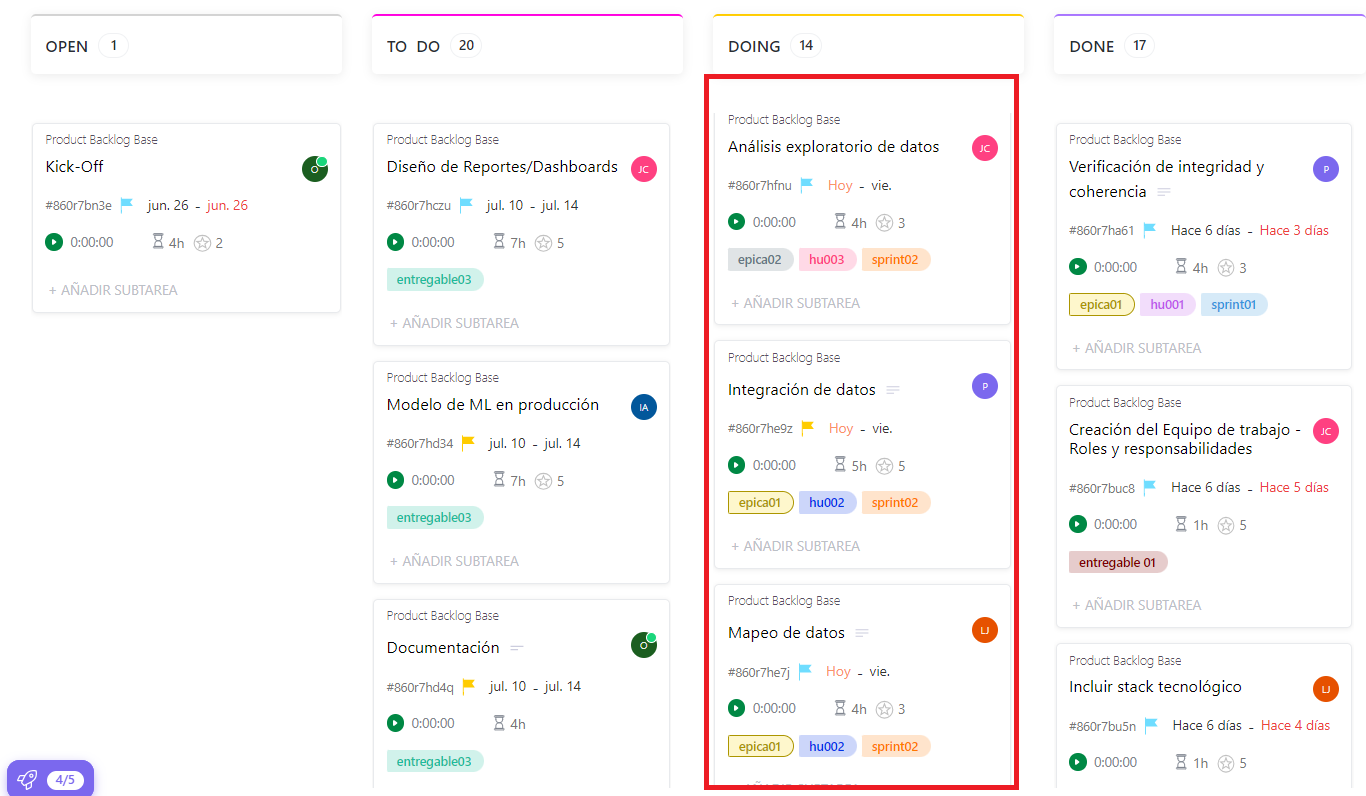
Descripción generada automáticamente con confianza bajaDiagrama

Descripción generada automáticamente

#### Product Backlog Refinado

El Product Backlog Refinado del Sprint 02 proporciona una visión clara de las tareas a realizar y establece una base sólida para el inicio del desarrollo durante el segundo sprint.

1. Product backlog -Sprint02



#### Incremento del Sprint

El incremento del Sprint actual se centra en la HU002, que implica la integración de datos y la creación de un manual de usuario para MongoDB. A continuación, se detalla el proceso de instalación de MongoDB y MongoDB Compass, junto con los pasos para configurar y activar el servidor en Windows.

Una vez completada la instalación, se mencionan los comandos básicos de MongoDB, como la visualización de colecciones y la búsqueda de información utilizando filtros. Además, se incluyen ejemplos de consultas SQL y su equivalente en MQL (MongoDB Query Language).

También se proporciona información sobre la actualización de datos, incluyendo la modificación de un único ID y la modificación de varios IDs simultáneamente.

Además, se explica el proceso de asignación de privilegios a los usuarios y la creación de colecciones, así como la carga de información utilizando MongoDB Compass.

Finalmente, se menciona la posibilidad de invocar la información desde Visual Studio Code (VSC) para trabajar con los datos de MongoDB.

En resumen, el incremento del Sprint se enfoca en la integración de datos y la creación de un manual de usuario para MongoDB. Se detallan los pasos de instalación, los comandos básicos, la actualización de datos, la asignación de privilegios, la creación de colecciones y la carga de información. También se menciona la opción de invocar la información desde VSC.

1. Incremento del Sprint-HU002

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

#### 3.5.2.1 Construcción de la HU002

La Historia de Usuario HU002 se centra en la construcción de la integración de datos. A continuación, se hará un resumen de los pasos clave involucrados en esta tarea:

#### Diseño adecuado del modelo

En esta etapa, se realiza un diseño cuidadoso de nuestro modelo que representa la estructura de la base de datos. Se identifican las entidades, atributos y relaciones entre ellas, y se definen las restricciones de integridad necesarias para garantizar la consistencia de los datos.

1. Diseño del modelo conceptual de datos

|  |  |  |
| --- | --- | --- |
| **N°** | **Colección** | **Descripción** |
| C001 | Calidad del Aire | La colección "Calidad del Aire" contiene información sobre la calidad del aire, centrándose en las moléculas detectadas por los sensores. Está estructurada por nueve columnas (llaves) que describen diferentes aspectos de la medición:  -Name: Nombre de la molécula medida. Por ejemplo, "Ozone (O3)".  -Measure: Tipo de medida realizada. Por ejemplo, "Mean" (media).  - Measure Info: Unidad de medida utilizada. Por ejemplo, "ppb" (partes por billón).  -Geo Type Name: Tipo de ubicación geográfica. Por ejemplo, "CD" (distrito censal).  -Geo Join ID: Identificador único de la ubicación geográfica. Por ejemplo, 313.  -Geo Place Name: Nombre del lugar geográfico. Por ejemplo, "Coney Island (CD13)".  -Time Period: Período de tiempo en el que se realizó la medición. Por ejemplo, "Summer 2013" (verano 2013).  -Start\_Date: Fecha y hora de inicio del período de medición. Por ejemplo, 2013-06-01T00:00:00.000+00:00.  -Data Value: Valor numérico de la medida. Por ejemplo, 34.64. |
| C002 | Combustion y CO2 | La colección "Combustion y CO2" detalla las emisiones de vehículos contaminantes de CO2, categorizados por modelo y marca. Está estructurada por las siguientes columnas:  - Model(Year): Año y modelo del vehículo (ejemplo: 2023).  - Make: Marca del vehículo (ejemplo: "Acura").  - Model: Modelo específico del vehículo.  - Vehicle Class: Clase del vehículo (ejemplo: "Full-size").  - Engine Size(L): Tamaño del motor en litros (ejemplo: 1.5).  - Cylinders: Número de cilindros del motor.  - Transmission: Tipo de transmisión del vehículo.  - Fuel (Type): Tipo de combustible utilizado (ejemplo: 4).  - Fuel Consumption(City (L/100 km)): Consumo de combustible en ciudad en litros por cada 100 km (ejemplo: 7.9).  - CO2 Emissions(g/km): Emisiones de CO2 en gramos por kilómetro (ejemplo: 167).  - CO2(Rating): Calificación de emisiones de CO2 (ejemplo: 6). |
| C003 | Contaminación sonora | En esta colección se detalla los diferentes sonidos producidos por un taxi que contaminan New york. Está estructurada por 6 columnas (llaves):  -Fecha: 2016-05-01T00:00:00.000+00:00 (formato de fecha y hora UTC).  -id\_borough: Identificador de la zona geográfica correspondiente al borough (distrito) donde se registraron los sonidos (ejemplo: 1)  -engine\_sounds: Calificación de los sonidos del motor (valor numérico). (ejemplo: 5)  -alert\_signal\_sounds: Calificación de los sonidos de las señales de alerta (valor numérico). (ejemplo: 3)  -total\_sounds: Calificación total de sonidos (valor numérico). (ejemplo: 8)  -borough\_name: Nombre del borough (distrito) donde se registraron los sonidos. (ejemplo: “manhattan”) |
| C004 | Taxis zonas | En esta colección se detallan las áreas y longitudes de los distritos y zonas de New York. Está estructurada por 6 columnas(llaves):  -Shape\_Leng: Longitud de la forma (ejemplo: 0.11635).  -Shape\_Area: Área de la forma (ejemplo: 0.00078).  -LocationID: Identificador de ubicación (ejemplo: 1). -  -Borough: Nombre del borough (distrito) correspondiente a la ubicación (ejemplo: “EWR”).  -Zone: Zona geográfica de la ubicación. (ejemplo:”Newark Airport”).  -service\_zone: Zona de servicio de la ubicación. (ejemplo:”EWR”) |
| C005 | Taxis rutas | En esta colección se detallan las rutas, tarifas, pasajeros y tipos de pago que fueron evaluados por un controlador y un taxímetro en la ciudad de New york. Está estructurado por 8 columnas (llaves):  -Fecha: Fecha del registro de datos (formato: "2022-01-01").  -Pasajeros por día: Número de pasajeros transportados durante el día.  -Viajes por día: Número total de viajes realizados durante el día.  -Tarifario por día: Monto total del tarifario registrado durante el día.  -Total recaudado por día: Monto total recaudado durante el día.  -Pago con tarjeta: Número de viajes pagados con tarjeta durante el día.  -Pago con efectivo: Número de viajes pagados en efectivo durante el día.  -Tipo de Taxi: Tipo de taxi utilizado para los viajes (ejemplo: "green"). |

#### Pipelines para alimentar el DW

Se crean los pipelines de extracción, transformación y carga (ETL) para obtener datos de diversas fuentes y alimentar el Data Warehouse (DW). Estos pipelines se encargan de recolectar, limpiar y preparar los datos antes de cargarlos en el DW.

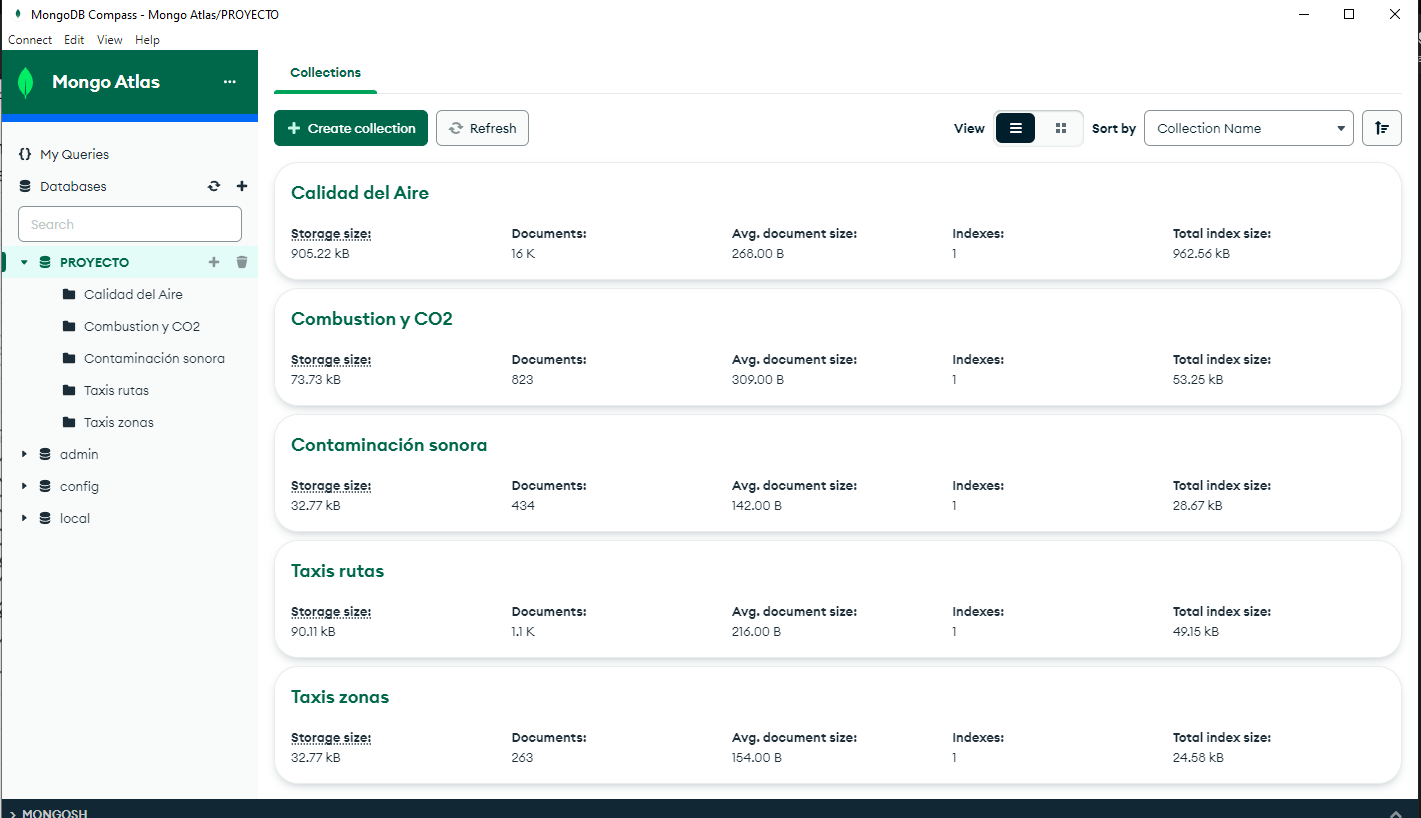
1. Detalle de los Pipelines

|  |  |  |
| --- | --- | --- |
| **N°** | **Pipelines** | **Descripción** |
| 01 | **Extracción** | Fuentes internet: Total Bruto: 50, Total Neto: 8  Archivos csv (17) ------------------------------------------------------------  Air\_Quality.csv  Light Duty Vehicles.csv  Alternative Fuel Vehicles US.csv  Electric and Alternative Fuel Charging Stations.csv  ElectricCarData\_Clean.csv  ElectricCarData\_Norm.csv  MY2012-2023 Plug-in Hybrid Electric Vehicles.csv  MY2012-2023 Battery Electric Vehicles.csv  annotations.csv  Clima2020-2023.csv  energy.csv  MY2023 Fuel Consumption Ratings.csv  Vehicle Fuel Economy Data.csv  data\_reports\_monthly-1.csv  df\_taxis.csv  taxi+\_zone\_lookup.csv  test\_taxis\_kaggle.csv |
| 02 | **Transformación** | Archivos Parquet (32) ------------------------------------------------------------  green\_tripdata\_2022-01.parquet  green\_tripdata\_2022-02.parquet  green\_tripdata\_2022-03.parquet  green\_tripdata\_2022-04.parquet  green\_tripdata\_2022-05.parquet  green\_tripdata\_2022-06.parquet  green\_tripdata\_2022-07.parquet  green\_tripdata\_2022-08.parquet  green\_tripdata\_2022-09.parquet  green\_tripdata\_2022-10.parquet  green\_tripdata\_2022-11.parquet  green\_tripdata\_2022-12.parquet  green\_tripdata\_2023-01.parquet  green\_tripdata\_2023-02.parquet  green\_tripdata\_2023-03.parquet  green\_tripdata\_2023-04.parquet  yellow\_tripdata\_2022-01.parquet  yellow\_tripdata\_2022-02.parquet  yellow\_tripdata\_2022-03.parquet  yellow\_tripdata\_2022-04.parquet  yellow\_tripdata\_2022-05.parquet  yellow\_tripdata\_2022-06.parquet  yellow\_tripdata\_2022-07.parquet  yellow\_tripdata\_2022-08.parquet  yellow\_tripdata\_2022-09.parquet  yellow\_tripdata\_2022-10.parquet  yellow\_tripdata\_2022-11.parquet  yellow\_tripdata\_2022-12.parquet  yellow\_tripdata\_2023-01.parquet  yellow\_tripdata\_2023-02.parquet  yellow\_tripdata\_2023-03.parquet  yellow\_tripdata\_2023-04.parquet  Archivos dbf (1)  taxi\_zones.dbf  Data final (3)  df\_taxis.csv  taxis\_zonas.csv  df\_conta\_sonora.csv  MY2023 Fuel Consumption Ratings.csv  Eliminación de nulos en las columnas y filas  Eliminación de columnas innecesarias  Renombramiento de los campos  Reestablecer los tipos de campos  Exportar JASON  Air\_Quality.csv  Eliminación de nulos en las columnas y filas  Eliminación de columnas innecesarias  Cambio de formato necesario Date  Reestablecer los tipos de campos  df\_taxis.csv  Eliminación de nulos en columnas y filas  Conversión de las fechas a formato yyyy-mm-dd  Eliminación de columnas innecesarias  Eliminación de valores atípicos  Renombrar los nombres de las columnas  taxis\_zonas.csv |
| 03 | **Carga** | **Eliminación de los valores nulos en filas y columnas**  **Merge de las zonas con distritos**  **Eliminación de columnas repetidas**  **Eliminación de valores atípicos**  df\_conta\_sonora.csv  **Eliminación de valores nulos en filas y columnas**  **Eliminación de valores atípicos**  **Conversión a formato yyyy-mm-dd**  **Ingesta en colleciones en Mongo Bd.json**  Archivos JSON (2) -----------------------------------------------------------------  ETL\_Air\_Quality.JSON  ETL\_combust\_MY2023.JSON  Archivos csv (5) ------------------------------------------------------------  taxis\_zonas.csv  df\_taxis.csv  df\_conta\_sonora.csv |

#### Data Warehouse

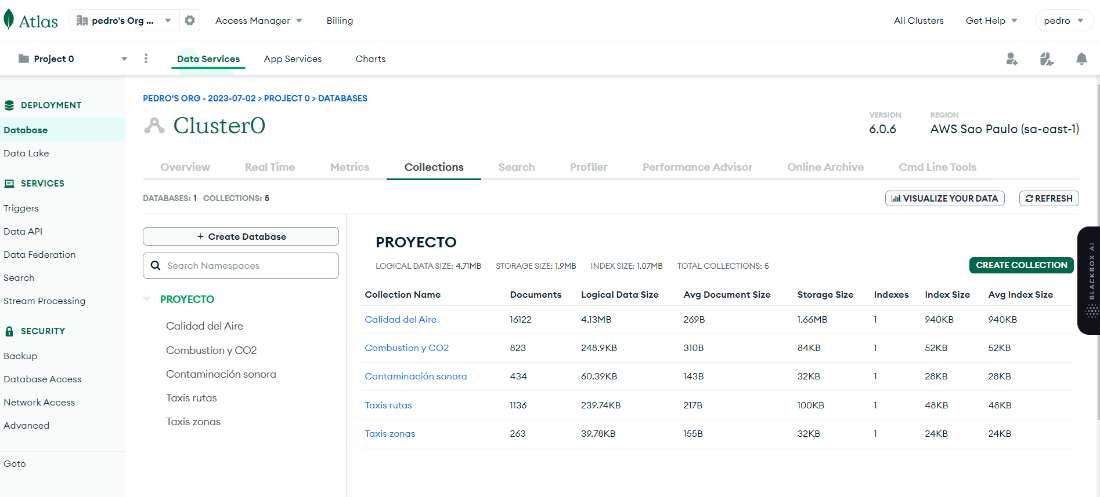
Aquí se almacenan los datos integrados y estructurados de manera que sean fácilmente accesibles y permitan realizar consultas y análisis eficientes.

1. Interfaz Mongo Compas



Nota. Aquí se muestra la interfaz de Mongo Compas en la cual se detalla las colecciones ingestadas en nuestro datawerehouse.

1. Servicio de Atlas en MongoDB



Nota. Proporciona una plataforma completamente administrada y escalable para almacenar y gestionar bases de datos MongoDB en la nube.

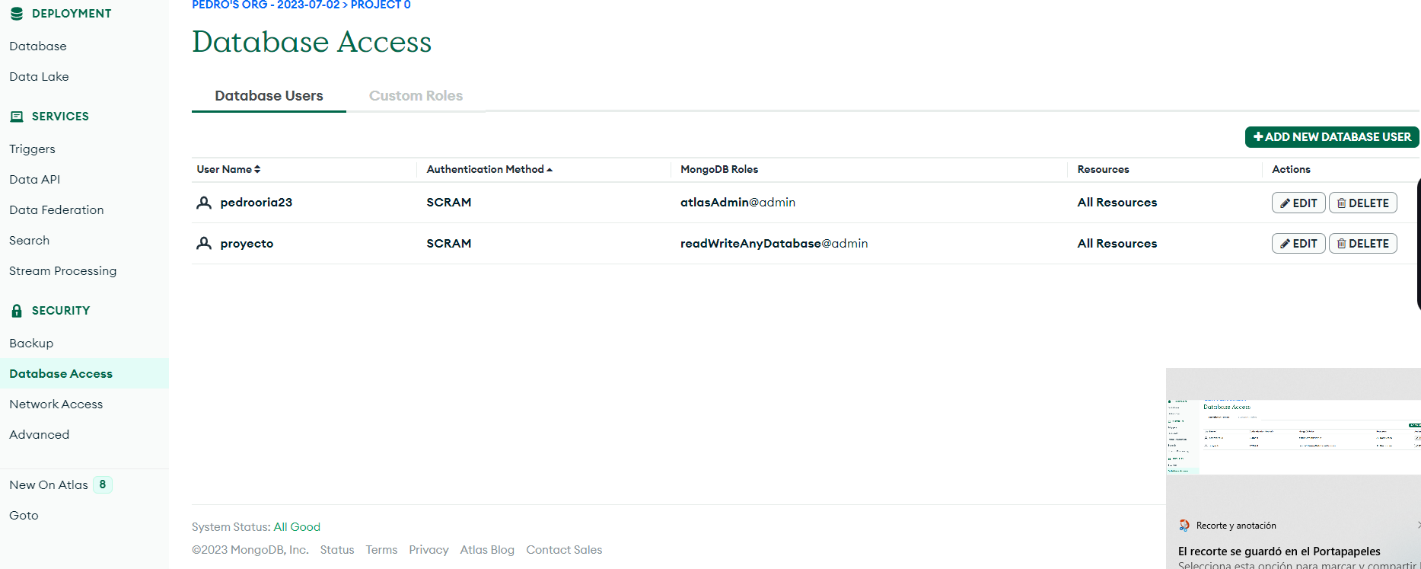
1. Diagrama

   Descripción generada automáticamente  
   Estructura del Data warehouse

#### Automatización

Monitoreo y notificaciones: Implementamos un sistema de monitoreo para verificar el estado y el rendimiento de los pipelines automatizados. Si ocurre algún error o se detecta un problema, se enviarían notificaciones automáticas a los responsables para que puedan tomar medidas correctivas de inmediato.

1. Creación de Usuarios



#### Validación de datos

Definir criterios de validación: Identifica los criterios que se utilizarán para verificar la calidad de los datos. Esto puede incluir reglas de negocio, restricciones de integridad, formatos esperados, rangos válidos, entre otros. Desarrolla scripts, consultas SQL u otras herramientas para automatizar las validaciones de datos. Esto te permitirá detectar errores y anomalías en los datos de manera eficiente y escalable.

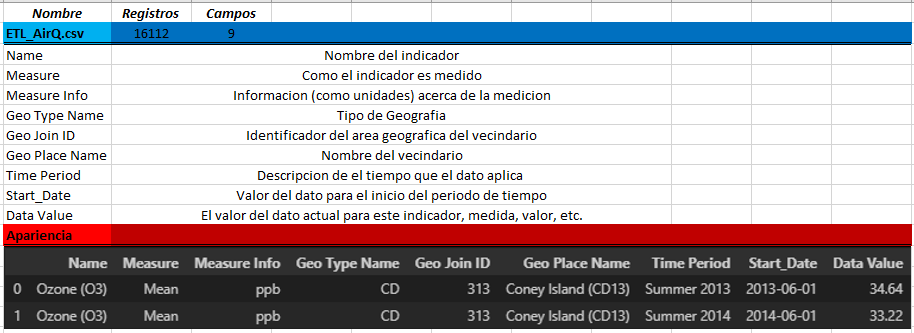
1. Tabla detalle de las Reglas de Negocio

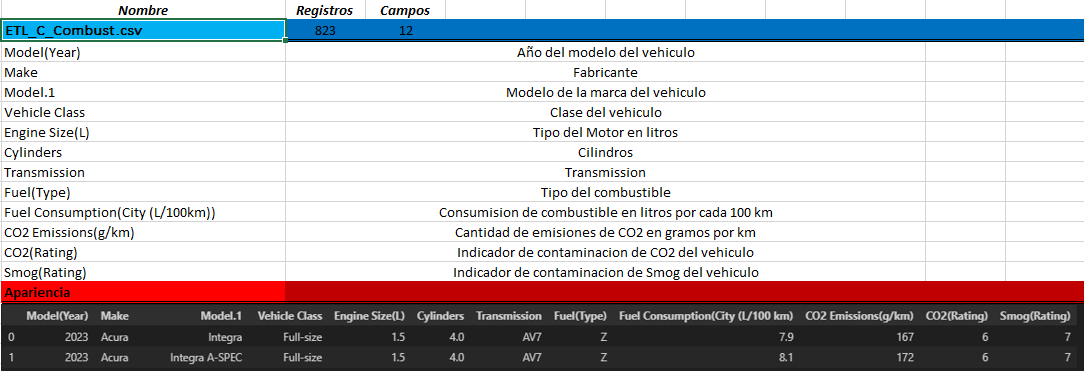
|  |  |  |  |
| --- | --- | --- | --- |
| **Regla Negocio** | **Descripción** | **Collection** | **Valor para el Negocio** |
| RN01 | En la colección Contaminación sonora, total\_sounds debe ser igual a la suma de engine\_sounds y alert\_signal\_sounds. | Contaminación sonora | Estas reglas verifican la coherencia de las mediciones del sonido. Esto es importante para evitar análisis erróneos basados en mediciones de sonido incorrectas. |
| RN02 | En la colección Combustión y CO2, Engine Size(L) no debe ser menor a 0. | Combustión y CO2 | Estas reglas aseguran que los datos de los vehículos y sus emisiones sean coherentes y lógicos. Esto es crucial para realizar análisis precisos y confiables sobre las emisiones de los vehículos.  Garantizar que los datos sean valores únicos |
| RN03 | En la colección combustión y CO2, CO2 Emissions(g/km) no puede ser menor a 0. | Combustión y CO2 |
| RN04 | En la colección Taxis zonas, LocationID debe ser un valor único. | Taxis zonas | Estas reglas aseguran que los datos sobre las zonas de taxis sean lógicos y únicos. Estos son fundamentales para evitar confusiones al analizar las zonas de taxis. |
| RN05 | Validar que las fechas se registren en un formato específico y que sean válidas (YYYY/MM/DD). | Taxis rutas | Esta regla es fundamental para el correcto seguimiento y análisis de los datos a lo largo del tiempo. Asegura que todas las fechas estén en un formato consistente, lo que facilita la realización de operaciones de comparación de fechas, análisis de series temporales y generación de informes periódicos. Además, la validación de la fecha también asegura que no se introduzcan datos erróneos en el sistema, lo que podría llevar a interpretaciones incorrectas y a decisiones de negocio erróneas. |
| RN06 | En la colección Taxis rutas, Pasajeros por día, Viajes por día, Tarifario por día, Total recaudado por día, Pago con tarjeta, Pago con efectivo no deben ser menor a 0. | Taxis rutas | Estas reglas verifican la coherencia de los datos de las rutas de taxis. Esta coherencia es crucial para realizar análisis correctos sobre los patrones de viaje y los ingresos de los taxis. |
| RN07 | Todos los formatos de datos ingestados deben estar en .json. | Calidad del Aire, combustión y CO2, Contaminación sonora, Taxis zonas, Taxis rutas | El valor principal de esta regla de negocio es la coherencia y la facilidad de uso. Los archivos JSON son un formato común para la transmisión de datos, especialmente en aplicaciones web, y son fáciles de leer y escribir tanto para las máquinas como para los humanos. Tener todos los datos en un solo formato facilita el procesamiento y el análisis, ya que no es necesario escribir y mantener diferentes rutinas de carga y transformación para cada formato de datos. También minimiza el riesgo de errores en la interpretación de los datos debido a diferencias en los formatos de archivo. |

1. Tabla detalle de la Validación de scripts de la regla del Negocio

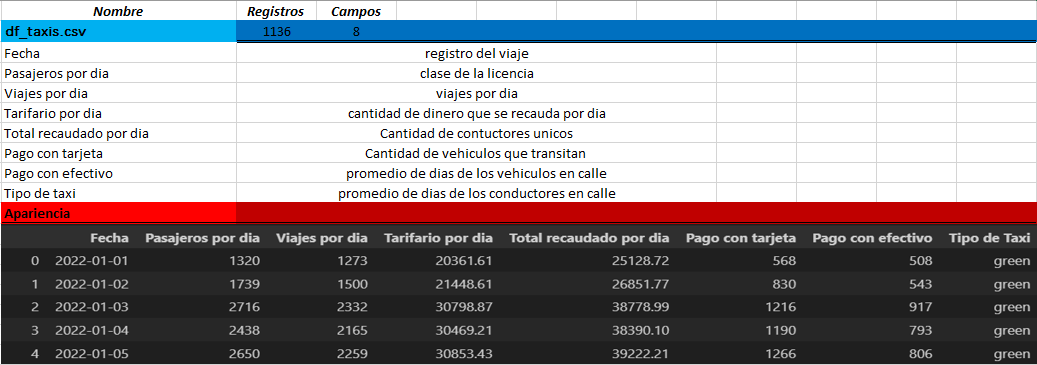
|  |  |
| --- | --- |
| **Regla Negocio** | **Ejecución del Script para su Validación** |
| RN01 | assert (df\_contaminacion\_sonora['engine\_sounds'] + df\_contaminacion\_sonora['alert\_signal\_sounds'] == df\_contaminacion\_sonora['total\_sounds']).all(), "Total sound score doesn't match the sum of engine and alert sounds" |
| RN02 | assert (df\_combustion\_co2['Engine Size(L)'] >= 0).all(), "Negative engine size found" |
| RN03 | assert (df\_combustion\_co2['CO2 Emissions(g/km)'] >= 0).all(), "Negative CO2 emissions found" |
| RN04 | assert (df\_contaminacion\_sonora[['engine\_sounds', 'alert\_signal\_sounds', 'total\_sounds']] >= 0).all().all(), "Negative sound score found" |
| RN05 | df = pd.read\_json('taxis\_rutas.json')  # Definir una función para comprobar si una fecha es válida y está en el formato correcto  def check\_date\_format(date\_str):  try:  pd.to\_datetime(date\_str, format='%Y/%m/%d')  return True  except ValueError:  return False  # Aplicar la función de comprobación a la columna de fecha  df['is\_date\_valid'] = df['Fecha'].apply(check\_date\_format)  # Verificar si hay alguna fecha no válida  invalid\_dates = df[~df['is\_date\_valid']]  print(invalid\_dates) |
| RN06 | assert (df\_taxis\_rutas[['Pasajeros por día', 'Viajes por día', 'Tarifario por día', 'Total recaudado por día', 'Pago con tarjeta', 'Pago con efectivo']] >= 0).all().all(), "Negative values found in Taxi Routes data" |
| RN07 | import pandas as pd  # Lista de colecciones  collections = ['calidad\_del\_aire.json', 'combustion\_y\_co2.json', 'contaminacion\_sonora.json', 'taxis\_zonas.json', 'taxis\_rutas.json']  # Definir una función para comprobar si un archivo es un JSON válido  def is\_valid\_json(file):  try:  pd.read\_json(file)  return True  except ValueError:  return False  # Comprobar cada archivo  for collection in collections:  if is\_valid\_json(collection):  print(f'{collection} es un archivo JSON válido.')  else:  print(f'{collection} no es un archivo JSON válido.') |

#### Diccionario de Datos

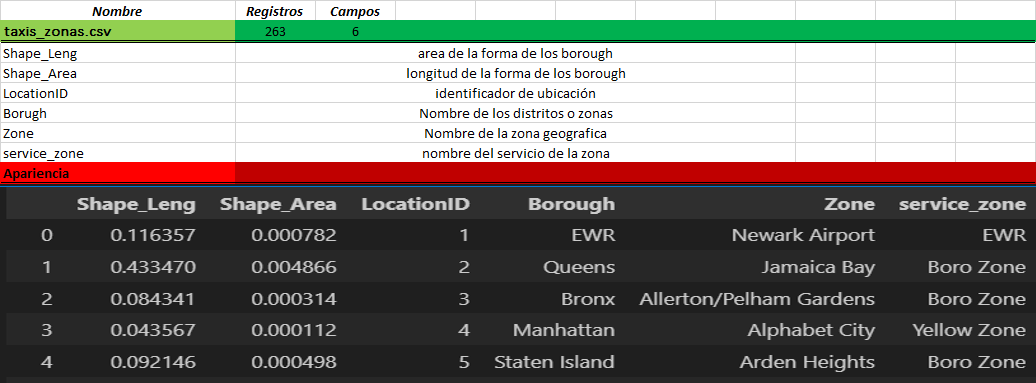
1.   
   Dataset ETL\_AirQ.csv
2. Dataset ETL\_C\_Combust.csv



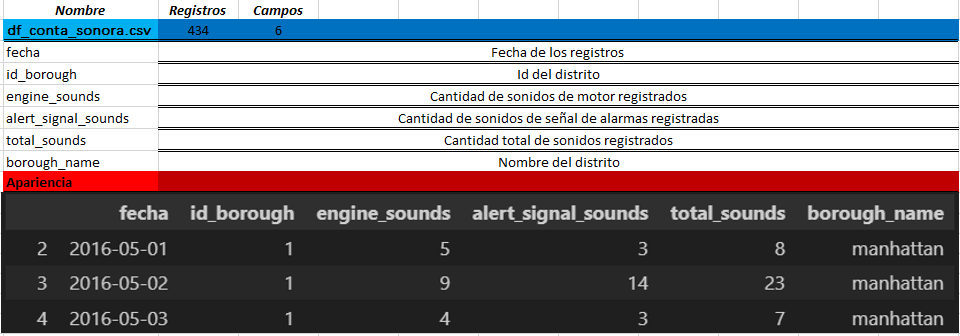
1. Dataset Taxis Rutas



1. Dataset Taxis Zonas



1. Dataset Sonora



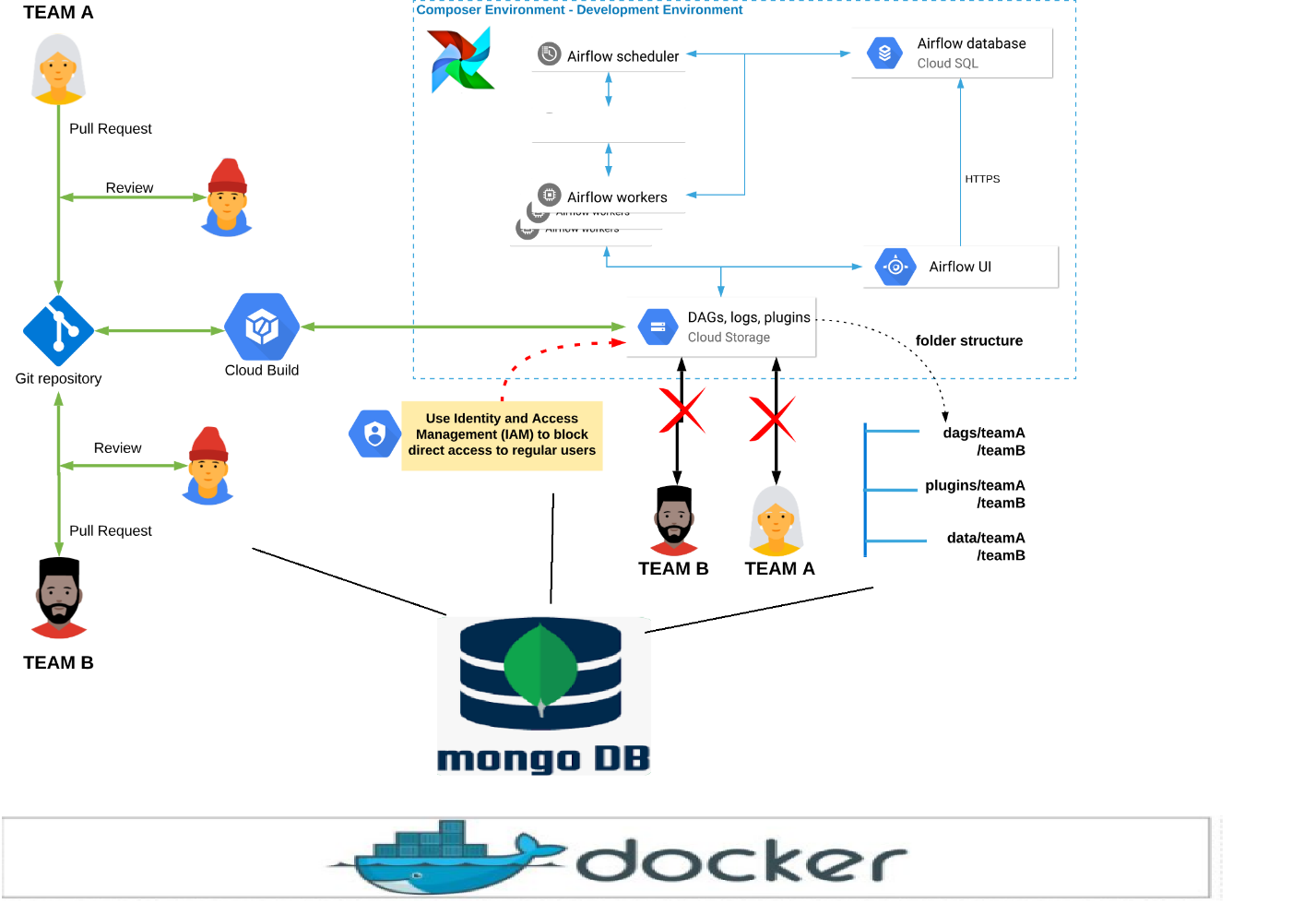
#### Workflow detallado en tecnologías

Docker: Utilizas Docker para crear y administrar contenedores virtuales que contienen todas las dependencias necesarias para ejecutar tu aplicación, incluyendo MongoDB y Airflow. Docker facilita la creación de entornos consistentes y reproducibles, lo que garantiza que tu aplicación se ejecute de la misma manera en diferentes entornos.

MongoDB: Utilizas MongoDB como tu base de datos para almacenar y gestionar tus datos. Puedes utilizar las capacidades de MongoDB para realizar operaciones CRUD (Create, Read, Update, Delete) en tus datos, así como para realizar consultas y agregaciones avanzadas. MongoDB te permite almacenar datos estructurados y no estructurados de forma flexible y escalable.

Airflow: Utilizas Apache Airflow como tu orquestador de tareas. Airflow te permite definir y programar flujos de trabajo complejos mediante la creación de DAGs (Directed Acyclic Graphs). Cada DAG consta de tareas individuales que se ejecutan en un orden determinado y pueden tener dependencias entre sí. Airflow te proporciona una interfaz fácil de usar para programar y monitorear tus flujos de trabajo, así como para gestionar el flujo de datos entre las tareas.

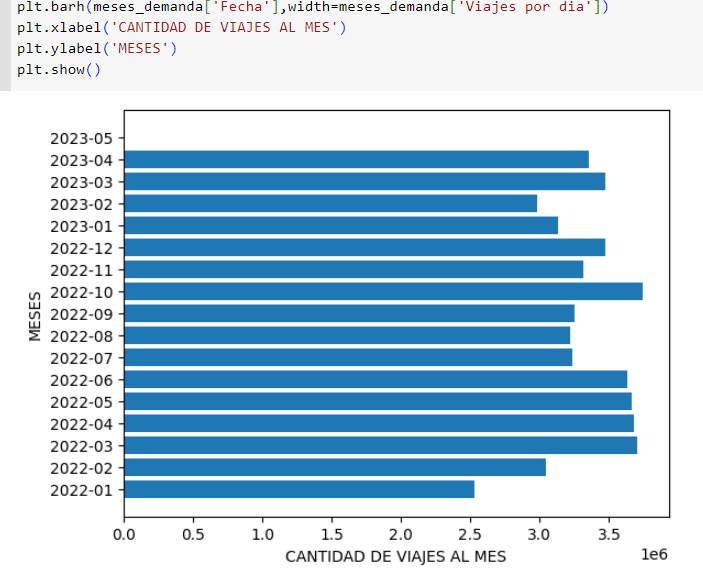
1. Workflow Detallado



#### 3.5.2.2 Construcción de la HU003

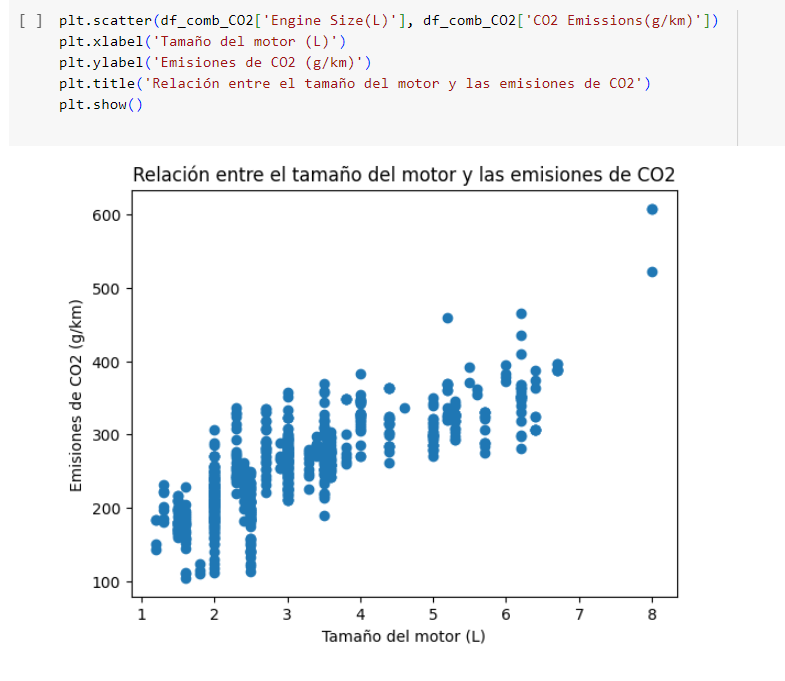
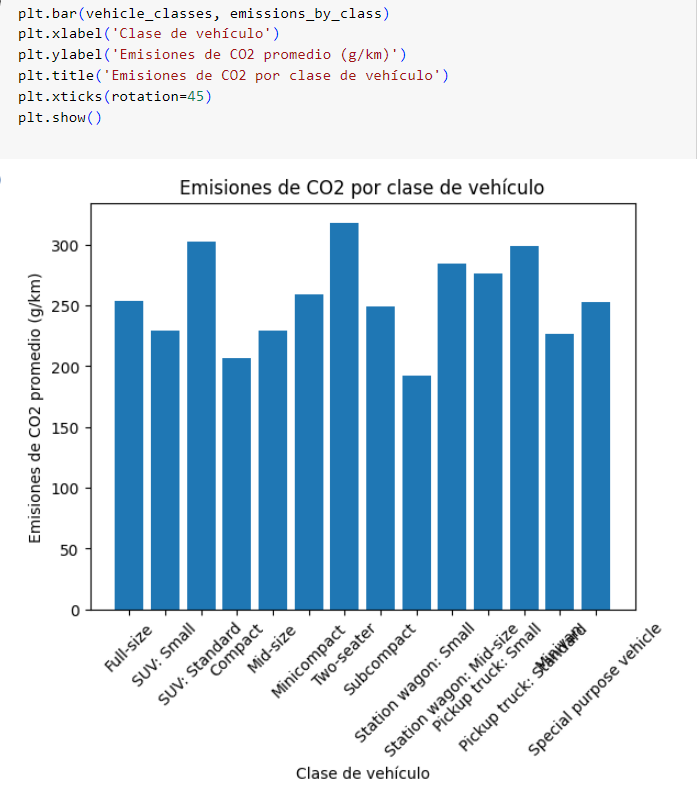
La Historia de Usuario HU003 se centra en el Análisis exploratorio de datos. A continuación, se hará un resumen de los pasos clave involucrados en esta tarea:

#### Análisis exploratorio de datos

1.   
   Gráfico de viajes por mes
2. Gráfico de correlación sobre las zonas de New York.

Gráfico, Gráfico de rectángulos

Descripción generada automáticamente

1.   
   Gráfico de dispersión de la tabla de Combustión y CO2.
2.   
   Gráfico de barras sobre las emisiones por vehículo en New york.
3. Gráfico, Gráfico de líneas

   Descripción generada automáticamente  
   Gráfico de Línea de los ingresos por mes.

#### Segmentación y agrupación

La segmentación consta de importar los datos y examinar su estructura, utilizando técnicas como clustering o segmentación basada en reglas para agrupar los datos en diferentes segmentos.

Analizar los segmentos: Examinar las características y comportamientos de cada segmento para comprender mejor las diferencias y similitudes entre ellos.

#### Identificación de insights

Este proceso consta de analizar relaciones y patrones realizando un análisis exploratorio más profundo para identificar relaciones y patrones interesantes entre variables,

Identificando valores atípicos para buscar excepciones que puedan proporcionar información relevante o insights únicos, realizando análisis comparativo en diferentes grupos o segmentos para identificar valores significativos y obtener insights adicionales, como por ejemplo la investigacion de features como Co2, contaminacion sonora, combustion y CO2.

#### Análisis estadístico

Los valores estadisticos se identificaron para las variables numericas validando la media, mediana y moda.

### 3.5.3 Daily Meeting

#### Gestión del tablero en el Sprint

* La frecuencia de las reuniones será de cinco veces por semana, de lunes a viernes.
* Cada reunión tendrá una duración máxima de 45 minutos.
* Durante la reunión se realizará la revisión de los avances presentes y futuros por cada historia de usuario.

Revisión del progreso. Se han realizado todas las historias de usuario con un progreso concluido del 100% tal como se especifica:

1. Sprint 02 – Revisión del progreso.

|  |  |  |
| --- | --- | --- |
| **Historia de Usuario** | **Progreso** | **Estado** |
| HU002 | 100% | Terminado |
| HU003 | 100% | Terminado |

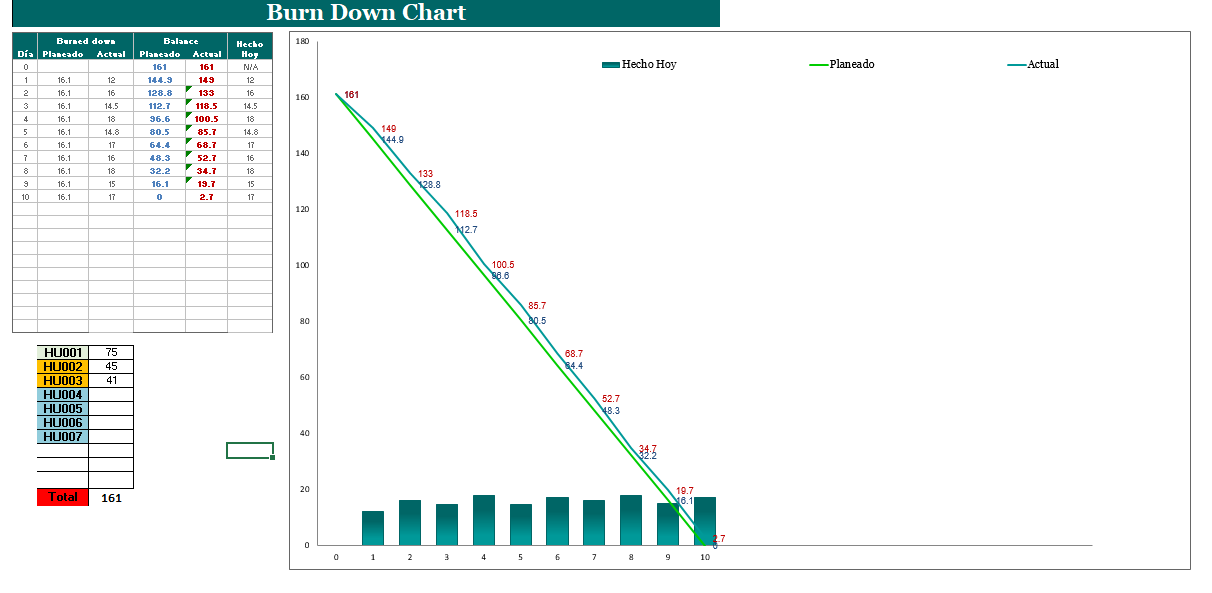
Identificación de impedimentos, bloqueos, dependencias y riesgos. Dentro de estas, se encuentran los siguientes impedimentos:

* Zonas horarias diferentes entre los desarrolladores del proyecto.

#### Sprint Burndown Chart

El gráfico de Burndown Chart se ingresa diariamente, después de cada dayli meeting y está realizado por el Scrum Master del proyecto con la finalidad de representar el esfuerzo esperado versus el real para el Sprint 1. Para poder obtener el porcentaje de avance diario, se toman los puntos de historia definidos en el Product Backlog, y se establece el esfuerzo diario obtenido.

1. Sprint Burdonwn Chart -Sprint 02



### 3.5.4 Sprint Retrospective

Dentro de los planes de mejora contemplados en la reunión realizada el día 07/07/2023, se obtuvieron los puntos detallados.

1. Imagen que contiene Interfaz de usuario gráfica

   Descripción generada automáticamente  
   Sprint Retrospective 02

# CAPITULO 7: CONCLUSIONES DEL PROYECTO

## 7.1 Conclusiones

Completar Aquí una conclusión por integrante De acuerdo con el objetivo general y específicos…

## 7.2 Recomendaciones

Completar Aquí una conclusión por integrante De acuerdo con el objetivo general y específicos…

# GLOSARIO

A continuación, se detalla la recopilación de definiciones o explicaciones de palabras que se mencionan en el presente documento:

1. Glosario de términos usados en el documento.

|  |  |
| --- | --- |
| **Término** | **Definición** |
| API | Interfaz de programación de aplicaciones constituido por conjuntos de funciones, procedimientos y subrutinas, que permite realizar la comunicación entre dos aplicaciones. |
| AWS | Conjunto de servicios de computación sobre la nube pública ofrecida por la empresa Amazon a través de internet. |
| Burndown chart | “Diagrama de Quemado” refiera a una representación gráfica del trabajo por hacer en un proyecto en el tiempo. |
| Deep learning | Campo de estudio del aprendizaje de máquinas (“Machine Learning”) enfocado en el uso de algoritmos inspirados en la  estructura y función del cerebro denominado Redes Neuronales Artificiales. |
| EDT | La Estructura de Desglose de Trabajo (EDT), representa una descomposición jerárquica orientada al producto entregable del  proyecto por ejecutar por el equipo de trabajo. |
| Machine learning | Campo de estudio que brinda a las computadoras la habilidad de aprender sin ser explícitamente programadas. Contempla la  programación de computadoras para que puedan aprender de una data de entrada. |
| Redes neuronales convolucionales. | Es un tipo de red neuronal artificial donde las neuronas artificiales, corresponden a campos receptivos de una manera muy similar a las neuronas en la corteza visual primaria de un cerebro biológico. |
| SCRUM | Marco de referencia empleada para el desarrollo ágil de un  software. |
| Sprint | Periodo de tiempo en el cual se realiza y evalúa un trabajo específico. |

# SIGLARIO

A continuación, se detalla la recopilación de los significados de las abreviaturas o que se mencionan en el presente documento:

1. Siglario de las siglas usadas en el documento.

|  |  |
| --- | --- |
| **Abreviatura** | **Significado** |
| AWS | Servicios de Amazon Web |
| BD | Base de Datos |
| CNN | Redes neuronales convolucionales. |
| EDT | Estructura de desglose de trabajo |
| HU | Historia de usuario |
| IDE | Entorno de desarrollo integrado |
| KPI | Indicador clave de desempeño |
| PMI | Asociación profesional de gestores de proyecto |
| TI | Tecnología de la información |
| UX | Experiencia de usuario |

# REFERENCIAS BIBLIOGRÁFICAS

* MongoDB. (s. f.). MongoDB Atlas: Cloud Document Database. <https://www.mongodb.com/cloud/atlas/lp/try4?utm_source=google&utm_campaign=search_gs_pl_evergreen_atlas_core_prosp-brand_gic-null_amers-cl_ps-all_desktop_eng_lead&utm_term=mongodb&utm_medium=cpc_paid_search&utm_ad=e&utm_ad_campaign_id=12212624314&adgroup=115749715423&cq_cmp=12212624314&gad=1&gclid=CjwKCAjwzJmlBhBBEiwAEJyLu5s7yvVRzR-Y23YVgsFUi1ktojXsvzxEvZ3NthwVYnbEpTZ2Y1_9eBoCvg8QAvD_BwE>
* AWS | Cloud Computing - Servicios de informática en la nube. (s. f.). Amazon Web Services, Inc. <https://aws.amazon.com/es/>
* Fundamentos de la metodología Agile. (s. f.). <https://www.wrike.com/es/project-management-guide/fundamentos-de-la-metodologia-agile/>
* Madedios. (2022b). Scrum: qué es y cómo funciona este marco de trabajo. www.wearemarketing.com. <https://www.wearemarketing.com/es/blog/metodologia-scrum-que-es-y-como-funciona.html>

# ANEXOS

## Gestión del Proyecto

1. Diccionario EDT, según sus entregables.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nombre | Descripción | Responsab les | Supuestos | Riesgos | Dependen cias |
| 1.2.11  Product Backlog | Permite hallar los principales requerimientos y requisitos para la construcción del aplicativo. | Responsable Principal: David Hospinal | Tener el Project Chárter terminado y aprobado | No se logre terminar el Project Chárter no se tendrá claro que roles son necesarios para el proyecto. | Project Charter, y EDT. |
| 1.3.1.1  Acta  de Sprint Planning | Sirve para inspeccionar el Backlog del Producto (Product Backlog) y que el equipo de desarrollo seleccione los Product Backlog Ítems en los que va a trabajar durante el siguiente Sprint. | Responsable Principal: David Hospinal | Tener el Product Backlog aprobado y priorizado según su esfuerzo y dimensiones. | No se logre terminar el Product Backlog aprobado y priorizado según su esfuerzo y dimensiones. Que son necesarios para el proyecto. | Project Charter y Product Backlog. |
| 1.3.1.2  Acta  de Sprint Backlog | Es la suma del Objetivo del Sprint, los elementos del Product Backlog elegidos para el Sprint, más un plan de acción | Responsable Principal: David Hospinal | Tener el Product Backlog aprobado y priorizado según su esfuerzo y dimensiones y el Sprint Planning. | No se logre terminar el Product Backlog y Sprint Planing. | Project Charter y Product Backlog y Sprint Planing. |
| 1.3.1.3  Acta  de Daily Scrum | Reunión para volver a planificar y tomar decisiones. | Responsable Principal: David Hospinal | Tener el Product Backlog aprobado y priorizado según su esfuerzo y dimensiones y el Sprint Planning y el Sprint Backlog. | No se logre terminar el Product Backlog y Sprint Planing y el Sprint Backlog. | Product Backlog y Sprint Planing y Sprint Backlog. |
| 1.3.1.4 Acta  de Sprint Retrospective | Último evento del Sprint y es la oportunidad para que el Equipo Scrum se analice a sí mismo y haga una propuesta de mejoras para que el desarrollo del siguiente Sprint sea más eficiente. | Responsable Principal: David Hospinal | Tener el Sprint Planning acabado. | No se logre terminar Sprint Planing . | Sprint Planing. |
| 1.3.1.5 Acta  de Sprint  Review. | reunión informal a la que asiste el equipo Scrum con el objetivo de ofrecer una demostración del prototipo del producto y determinar qué pendientes fueron terminados. | Responsable Principal: David Hospinal | Tener el Sprint Retrospective acabado y prototipos con su criterio de aceptación. | No se logre hacer el Sprint Retrospective los prototipos. | Sprint Retrospective, prototipos con su criterio de aceptaión. |
| 1.4.1 Tablero  Kanban | Herramienta ágil de gestión de proyectos diseñada para ayudar a visualizar el trabajo, limitar el trabajo en curso y maximizar la eficiencia (o el flujo). Puede ayudar tanto a los equipos ágiles como a los de DevOps a definir el orden de su trabajo diario. | Responsable Principal: David Hospinal | Tener la gestión de cronograma y EDT acabado. | No se logre hacer el cronograma. | Cronograma. |
| 1.4.2 Burndown  Chart | Representación gráfica del trabajo por hacer en un proyecto en el tiempo. Usualmente el trabajo remanente se muestra en el eje vertical y el tiempo en el eje horizontal. Es decir, el diagrama representa una serie temporal del trabajo pendiente. | Responsable Principal: David Hospinal | Tener los Sprint y product backlog y cronograma del proyecto acabado. | No se logre hacer el cronograma y tablero Kanban. | Cronograma y tablero Kanban. |
| 1.5.3.2 Entrega  del producto final | Es la entrega Final del producto | Responsable Principal: David Hospinal | Tener el modelo terminado y validado. | No se logre testear el modelo | Validaciones y test |