



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Informática

Aplicación para asignar beneficios a clientes de una
compañía de seguros

Trabajo Fin de Grado

Grado en Ingeniería Informática

AUTOR/A: Ruiz Mico, Miguel

Tutor/a: Vendrell Vidal, Eduardo

CURSO ACADÉMICO: 2021/2022



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Aplicación de asignación de beneficios para los clientes de una aseguradora

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: Miguel Ruiz Micó

Tutor: Vendrell Vidal, Eduardo

Curso: Curso 2021/2022

Resumen en castellano

Este proyecto consiste en una aplicación realizada para una empresa ficticia de seguros de vida, la cual ha encomendado que realicemos una aplicación que otorgue descuentos o beneficios a los clientes al realizar retos que estén relacionados con actividades físicas para mejorar la salud. De esta manera, aparte de como método de promoción para la empresa para atraer clientes, el cliente se siente más satisfecho por el hecho de poder obtener diversos beneficios y la aseguradora hace que sus clientes se encuentren en un estado más saludable, lo que hará que haya menor riesgo de que surjan problemas de salud.

Para ello la empresa se propone medir la actividad física que realiza el cliente mediante algún tipo de pulsera o reloj que le proporciona. A partir de la información que va recolectando el dispositivo que mide la que actividad física y junto a los datos personales del cliente (nombre, edad, altura, ...) como también la que. El proyecto aborda desde la recogida de datos hasta el desarrollo de una aplicación para los clientes.

En una primera fase se aborda la recogida y tratamiento de los datos. Para ello se crea una base de datos donde se vuelcan y almacenan todos los datos que se reciben. Posteriormente, se desarrolla una aplicación para que los clientes puedan visualizar sus rendimientos, al tiempo que se analizan los datos para que la empresa pueda tomar decisiones sobre este seguimiento. En la primera parte se ven y aplican algunas técnicas relacionadas con el big data y el data engineering, que permiten transportar los datos desde donde se obtienen hasta la base de datos, a la vez que también se van procesando y transformando de forma que queden limpios y organizados.

En la segunda parte, se utilizan herramientas de Business Intelligence que permiten sacar el máximo partido a los datos y realizar informes sobre la aplicación que resulten completos, claros y concisos. Se trata de la parte de análisis, dedicada a la compañía (B2B). Por el otro lado está la parte dedicada a los clientes (B2C), en la que se crea la aplicación y se aplican las distintas tecnologías que hacen uso de la base de datos hasta la interfaz de usuario con la que puede interactuar el cliente.

Resumen en valenciano

Aquest projecte consisteix en una aplicació realitzada per a una empresa fictícia d'assegurances de vida, la qual ha encomanat que realitzem una aplicació que atorgue descomptes o beneficis als clients en realitzar reptes que estiguen relacionats amb activitats físiques per a millorar la salut. D'aquesta manera, a part de com a mètode de promoció per a l'empresa per a atraure clients, el client se sent més satisfet pel fet de poder obtindre diversos beneficis i l'asseguradora fa que els seus clients es troben en un estat més saludable, la qual cosa farà que hi haja menor risc que sorgisquen problemes de salut.

Per a això l'empresa es proposa mesurar l'activitat física que realitza el client mitjançant alguna mena de polsera o rellotge que li proporciona. A partir de la informació que va recollint el dispositiu que mesura la que activitat física i al costat de les dades personals del client (nom, edat, altura, ...) com també la que. El projecte aborda des de la recollida de dades fins al desenvolupament d'una aplicació per als clients.

En una primera fase s'aborda la recollida i tractament de les dades. Per a això es crea una base de dades on es bolcaran i emmagatzemaran totes les dades que es reben. Posteriorment, es desenvolupa una aplicació perquè els clients puguen visualitzar els seus rendiments, al mateix temps que s'analitzen les dades perquè l'empresa puga prendre decisions sobre aquest seguiment. En la primera part es veuen i apliquen algunes tècniques relacionades amb el *big data i el data *engineering, que permeten transportar les dades des d'on s'obtenen fins a la base de dades, alhora que també es van processant i transformant de manera que queden nets i organitzats.

En la segona part, s'utilitzen eines de *Business *Intelligence que permeten traure el màxim partit a les dades i realitzar informes sobre l'aplicació que resulten complets, clars i concisos. Es tracta de la part d'anàlisi, dedicada a la companyia (B2B). Per l'altre costat està la part dedicada als clients (B2C), en la qual es crea l'aplicació i s'apliquen les diferents tecnologies que fan ús de la base de dades fins a la interfície d'usuari amb la qual pot interactuar el client.

Resumen en inglés

This project consists of an application made for a fictitious life insurance company, which has commissioned us to make an application that grants discounts or benefits to customers when they perform challenges that are related to physical activities to improve health. In this way, apart from as a method of promotion for the company to attract clients, the client feels more satisfied by the fact of being able to obtain various benefits and the insurer makes its clients be in a healthier state, which will make There is less risk of health problems.

For this, the company intends to measure the physical activity carried out by the client through some type of bracelet or watch that it provides. From the information that is collected by the device that measures what physical activity and together with the client's personal data (name, age, height, ...) as well as what. The project deals with everything from data collection to the development of an application for customers.

In a first phase, the collection and processing of data is addressed. To do this, a database is created where all the data received is dumped and stored. Subsequently, an application is developed so that customers can view their performance, while the data is analyzed so that the company can make decisions about this monitoring. In the first part, some techniques related to big data and data engineering are seen and applied, which allow the data to be transported from where it is obtained to the database, while also being processed and transformed so that it is clean. and organized.

In the second part, Business Intelligence tools are used to make the most of the data and make reports on the application that are complete, clear and concise. This is the analysis part, dedicated to the company (B2B). On the other hand, there is the part dedicated to clients (B2C), in which the application is created and the different technologies that make use of the database are applied to the user interface with which the client can interact.

Índice

Contenido

1. Introducción.....	7
1.1 Objetivos.....	7
1.2 Estructura.....	8
2. Análisis del problema	10
2.1 Solución propuesta.....	10
3. Diseño de la solución	13
4.1 Arquitectura del sistema	13
4.3 Tecnología utilizada.....	19
4. Desarrollo de la solución propuesta.....	22
4.1 Fuente de datos	23
4.2 Base de datos	23
4.3 Procesamiento de datos	25
4.4 Análisis de datos.....	28
5. Implantación	33
6. Conclusiones.....	34
7. Trabajos futuros.....	36
8. Glosario	37
9. Referencias	40

Ilustraciones

Ilustración 1 - Herramientas de mensajería de datos	11
Ilustración 2-Diseño de base de datos relacional	12
Ilustración 3 -Arquitectura del sistema	14
Ilustración 4 - Diseño relacional de la base de datos.....	17
Ilustración 5- Sistema de comunicación de kafka.....	18
Ilustración 6-Logo de Docker	19
Ilustración 7 -Logo de kafka.....	20
Ilustración 8- Logo de python	20
Ilustración 9 - Logo de MySQL	20
Ilustración 10 - Logo de PowerBI	21
Ilustración 11- Imagen de bubble	21
Ilustración 12 - Contenedores lanzados en docker.....	22
Ilustración 13-Obtención de los datos de imágenes.....	23
Ilustración 14-Página inicial de MySQL	24
Ilustración 15-Estructura de la base de datos	25
Ilustración 16- Creación del producer en python	25
Ilustración 17- Creación del consumer en python.....	25
Ilustración 18 – Fórmula de semiverseno	26
Ilustración 19- Calorías quemadas según el ejercicio	26
Ilustración 20-Procedimiento para actualizar tabla workout.....	27
Ilustración 21-Conectar PowerBi con la base de datos.....	28
Ilustración 22-Indicador de obesidad	29
Ilustración 23-Dashboard 1	29
Ilustración 24 - Dashboard 2	30
Ilustración 25- Conexión a la base de datos.....	31
Ilustración 26 - Query acerca de la información diaria	31
Ilustración 27 - Página de la aplicación	32

1. Introducción

En la industria globalizada en la cual nos encontramos, está ocurriendo un cambio de época la cual exige la aparición de nuevas tecnologías que faciliten el dinamismo en la confección de estrategias y en la forma de llevar a cabo los procesos. Este escenario genera el surgimiento de tecnologías, como el tratamiento y análisis masivo de datos que favorecerá a la creación de nuevas herramientas capaces de incorporar a la empresa el atributo de innovación.

Este proyecto se basa en la construcción de un sistema que otorgue beneficios a los clientes pertenecientes a una aseguradora con tal de que ambos salgan beneficiados, por parte de los clientes porque obtienen una serie de puntos que luego podrán canjear por una serie de bonificaciones que aportara la aseguradora.

Esto genera un mayor grado de satisfacción, a parte de la satisfacción de los clientes, cosa que es importante ya no solo para consolidarlos, también para la captación de nuevos clientes y de la imagen de marca de la empresa, sino que también mejoraría la salud de dichos clientes, cosa que generaría que sufran menos riesgos de sufrir enfermedades o problemas de salud.

En este proyecto se ofrecen dos productos diferentes en este proyecto, por un lado, la para los clientes (B2C) una aplicación con la cual se pueden conectar para ver los clientes, y por otra parte, un producto más dirigido hacia la empresa (B2B), que es un estudio acerca de los clientes y la realidad física que realizan

La aplicación permite al usuario realizar un seguimiento, pudiendo ver cuál es su rendimiento, además de poder ver los distintos retos con los objetivos que hay que cumplir y los beneficios que se obtienen en caso de lograrlo; y por otro lado se encuentra el estudio que se hace, en el cual se reporta la información de los distintos usuarios para observar su evolución y desarrollo para poder ver la utilidad que tiene el sistema de beneficios creado, además de poder estudiar a los el estado físico de los clientes y su estado de salud

Durante este proyecto podremos observar la importancia que pueden llegar a tener, tanto los datos por sí mismos, como la gestión y tratamiento que realizas sobre ellos para llegar a obtener información, y realizar productos únicos para la gente.

1.1 Objetivos

La finalidad de este proyecto consiste en la construcción de una solución que proporcione información relevante acerca de un grupo de estudio en específico mediante un análisis sobre los datos se proporcionan sobre ellos apoyándose sobre todo en el conocimiento de las tecnologías de la información y en concreto en las técnicas y herramientas que aporta el Business Intelligence. En concreto, mediante la construcción de una serie de graficas que ayudaran a relevar información sobre el objeto de estudio, establecidos en un cuadro de mando, también conocido como dashboard.

Por otro lado, también tiene el objetivo de promover este tipo de tecnologías y las virtudes que tiene las soluciones de Business Intelligence y como estas pueden resultar de utilidad y ayuda a las organizaciones

Para llegar a cumplir con el propósito principal del trabajo, dicho propósito será dividido en objetivos específicos. La mayoría de estos objetivos están interrelacionados y localizados a lo largo del proyecto. Por esto es por lo que el cumplimiento de los objetivos parciales establecidos ayudase al logro del fin general del proyecto. El conjunto de objetivos específicos es mostrado a continuación

- Desarrollar una estructura física capaz de establecer, concretar y normalizar toda la información
- Saber identificar la información que pueda resultar útil para la empresa a la hora de la toma de decisiones
- Diseñar y desarrollar un sistema de inteligencia de negocio, que permita un análisis correcto de los datos recogidos de forma más sencilla e intuitiva.
- Construir una interfaz intuitiva y funcional para los usuarios
- Ser capaz de gestionar la información en tiempo real

1.2 Estructura

A la hora de realizar un estudio, es importante establecer una serie de pautas y procedimientos a seguir para así obtener resultados más correctos y precisos. En este caso se han establecido una serie de puntos conforme a los objetivos establecidos anteriormente.

Para empezar, se ha sido detallado el contexto del trabajo y descrito la situación que envuelve a todas estas tecnologías y herramientas que han sido relacionadas con el Business Intelligence. Se estudiará todos los conceptos que se encuentren relacionado con ellos, tanto como la arquitectura, el desarrollo de nuevas herramientas, los componentes hardware que han ido surgiendo los últimos años, ... Nos centraremos en las distintas metodologías que han sido diseñadas para llevar a cabo este tipo de proyectos y poder aportar una solución al problema planteado de la manera más eficaz y eficiente.

Para empezar, se hace un breve análisis del problema que surge y al cual se enfrenta este proyecto, donde a partir de la identificación de requisitos, se dará solución a los objetivos planteados. Establecidos los requisitos, se realiza un estudio de distintos caminos a seguir y distintas tecnologías con las que se puede trabajar. Sopesando todas las opciones, se crea la mejor solución posible, la cual tiene que soportar los requisitos establecidos

Con todas las decisiones tomadas se realiza el diseño de la solución donde se ve las partes por las que va a estar compuesta la solución que se ha decidido aplicar, y donde se explicaran en que consiste cada una de las partes y que se tarea se realiza en cada una de ellas, dando una explicación detallada en algunas de las partes para comprender mejor el funcionamiento del sistema, y comentando cuales son y que hacen las tecnologías aplicadas a cada parte de la arquitectura.

Más adelante, se procede a desarrollar esta solución, en este apartado se ira viendo como se ha realizado cada una de las partes del proyecto, y se enseñaran los pasos que se han ido realizando para elaborar la solución hasta llegar al producto final.

Llevado a cabo el sistema y habiéndolo implementado, es hora de extraer conclusiones sobre el proyecto, en ella se hace autocrítica acerca de los resultados obtenidos, y se hace una comparación con los objetivos establecidos al principio, viendo si estos han sido satisfechos, dando una explicación en cada uno de porque si o porque no.

Terminado el proyecto, se hace un apartado de trabajos futuros. Esta parte habla sobre mejoras y cambios que se puede aplicar al proyecto que se ha realizado o de otros proyectos nuevos que pueden surgir a partir de las herramientas y tecnologías que se han visto.

Al final del todo aparece un glosario donde aparecen términos que pueden resultar desconocidos para el lector y un apartado de referencias donde aparece las referencias bibliográficas a las que se ha recurrido para obtener información del proyecto.

2. Análisis del problema

Con este proyecto se quiere realizar un producto que resulte interesante y atractivo. Cada día las tecnologías van evolucionando más y las empresas requieren de innovaciones en sus productos para generar interés en el mercado y atraer clientes, buscando sobre todo en el apartado tecnológico dichos beneficios. Para ofrecer este tipo de servicios es necesario realizar muchos procesos y, por tanto, hace que se vuelva complicado crear un sistema que contengan todos los procesos y todos funcionen correctamente, y no surjan errores por el camino.

Estas aplicaciones dependen totalmente de los datos que se reciben de los clientes en todo momento, por ello el problema uno de los problemas principales que surge es la gestión de los datos que estamos recibiendo para que lleguen correctamente hasta la aplicación y además de tenerlos disponibles para el análisis que se presenta a la empresa.

Para conseguir esto q se ha establecido una serie de requisitos que habría que realizar para poder cumplir con los objetivos que se han preestablecido anteriormente

- Tener un sistema que sea capaz de gestionar los mensajes que se lanzan en streaming
- Crear una base de datos para almacenar la información
- Disponer de las herramientas que permitan realizar informes mediante graficas
- Aplicar operaciones sobre los datos para obtener la información deseada
- Conectar la base de datos con las distintas herramientas.
- Adquirir datos realistas aplicables al proyecto

A partir de aquí se hace un análisis más profundo sobre los requerimientos que hay que cumplir y que tareas hay que realizar para el proyecto se lleve a cabo correctamente y obtengamos la mejor solución posibles, para ello se hace una investigación acerca de este tipo de proyectos y cómo se podría resolver cada uno de los distintos requisitos planteados. Habrá que mirar no solamente los procesos que se deben realizar sino las tecnologías necesarias para poder implementar dichas soluciones.

2.1 Solución propuesta

El proyecto se divide en 3 partes diferenciadas, pero antes de explicar cada una de las partes hay que indicar de donde se va a extraer la información, una buena fuente de datos de donde poder sacar muchos datos reales es la plataforma de kaggle, plataforma donde se almacenan csv con caso de información real, esta sería una fuente importante si los datos a utilizar fuesen ya predefinidos, pero para el proyecto se necesitan datos que se van actualizando con el tiempo, por tanto, no nos sirve. Como es muy difícil obtener información de tantos usuarios diferentes reales, se ha decidido realizar una demo con datos simulados que se asemejen lo

máximo posible a la realidad, para ello se hace una investigación de cómo funcionan los dispositivos y los valores que pueden alcanzar las distintas variables que tengamos.

Una vez explicada esta parte, se va con la primera parte que constara del tratamiento que se le hace al dato, es decir, se podrá ver de dónde se han obtenido los datos y de qué manera se consigue moverlos entre los diferentes sistemas, y las transformaciones que realizamos. En Business Intelligence se suele utilizar herramientas de tipo ETL. Estas herramientas son muy utilizadas hoy en día para la limpieza y procesamiento del dato, sin embargo estas tecnologías son más adecuadas para tareas de análisis predictivo o minería de datos, además, no se podría aplicar, ya que suelen ser herramientas diseñadas para realizar procesamientos con datos estáticos, es decir, utilizar datos que no varían con el tiempo, para la aplicación se necesita constantemente realizar envíos y modificaciones conforme los datos van siendo lanzados, así que es necesario una o varias herramientas para realizar procesamiento de datos en streaming que permita gestionar los volúmenes de datos que van llegando en tiempo real. Entonces se ha dividido la tarea en dos partes por un lado se encuentra el proceso de ingesta de datos que se encargara de realizar los envíos de información y gestionar los datos que van llegando, y por otro lado se necesita alguna tecnología para realizar las modificaciones sobre los datos y cargarlos en la base de datos



Ilustración 1 - Herramientas de mensajería de datos

En la segunda parte estaría el apartado de la realización de la estructura física donde almacenar los datos, hay proyectos donde directamente se conecta la herramienta de procesamiento con el producto final, en este caso ha decidido no hacerse así debido a los beneficios que te aporta crear una base de datos. Utilizar una base de datos, te permite almacenar toda la información en un solo lugar, fácil, además de mantener la información que se ha ido procesando en caso de que se produzca un fallo en el sistema en la parte, facilita la conexión con diferentes sistemas.

Luego dentro del tipo de bases de datos, existen dos tipos diferentes las bases de datos: relacionales y las no relacionales, en este proyecto se utilizará una base de datos relacional, esta decisión se ha tomado debido a que la sencillez que tiene trabajar en ella, a que es fácil de crear y muy accesible con la opción de ampliar la base de datos sin perjudicar a las aplicaciones existentes. Además de crear una estructura definida que facilita el hecho de realizar consultas sobre la base de datos y nos garantiza la uniformidad de los datos y sin olvidar que en él no existe la duplicidad de registros. Ya por último aporta atomicidad que permite establecer reglas y normas sobre los registros de las distintas tablas. La base de datos relacional más usada y conocida es MySQL junto con Oracle, seguida por SQL Server y PostgreSQL, entre otras.

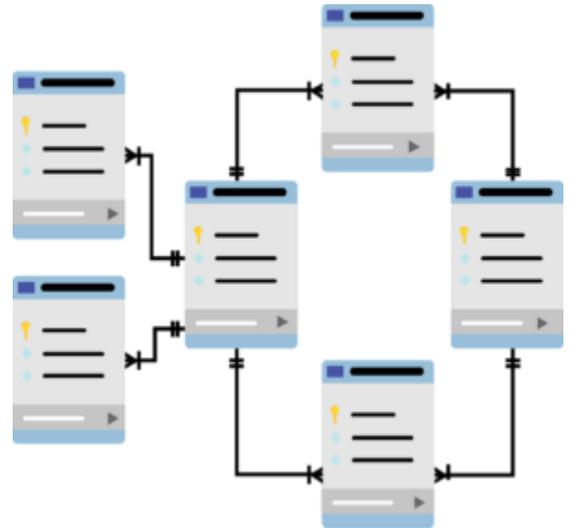


Ilustración 2-Diseño de base de datos relacional

Luego en la parte final ya entraría los dos servicios que se quieren proporcionar. La parte más dirigida al negocio de análisis de la empresa se utilizan técnicas de análisis descriptivo, ya que es el tipo de análisis que se adecua más a lo que se quiere obtener. Para realizar los informes con este tipo de analítica, hay que recurrir a herramientas de Business Intelligence que permitan este tipo de análisis como puede ser PowerBi o Tableau.

Luego por otro lado está la aplicación. Aquí existen muchas maneras para crear aplicaciones ya que, hoy en día hay una gran variabilidad de herramientas para crear aplicaciones. Ya que existen muchas herramientas de low-code que aportan funcionalidades y que facilitan la labor de creación de interfaces, para que resulten más intuitivas y atractivas visualmente, se ha decidido utilizar algunas de estas herramientas para crear la aplicación deseada.

Para la mejora del sistema también algunas de las tecnologías que se van a utilizar se van a desplegar mediante Docker, ya que es una herramienta que permite el despliegue de aplicaciones mediante contenedores en otros servidores, esto te otorga una gran cantidad de beneficios como puede ser en temas de seguridad, el rápido despliegue de las aplicaciones en segundos y el aislamiento entre ellas.

3. Diseño de la solución

Entrando dentro del diseño de la solución, se va a ver el diseño completo del sistema que se va a construir con sus distintos componentes y tecnologías. Además, se explica detalladamente en que consiste cada parte del sistema para que quede entendido correctamente su funcionamiento, tanto lo que hace cada uno de los procesos, como que hacen cada una de las tecnologías que se usan y porque se han elegido esas tecnologías; y también cosas que serán necesarias predefinir con anterioridad para aplicarlas a los procesos

Este proyecto, aunque no en vaya a albergar distintos servicios en contenedores Docker como se explicara más detalladamente más en adelante, pero muchas partes de los servicios sí que se van a ser lanzados en local. Por eso, se considera importante destacar las características del equipo donde se va a trabajar:

- CPU: Intel(R) Core (TM) i7-10870H CPU @ 2.20GHz, 2208 Mhz, 8 procesadores principales, 16 procesadores lógicos
- RAM: 16,00 GB
- Sistema operativo: Windows 10 Home 64 bits (10.0, compilación 19044)
- Modelo del sistema: ASUS TUF Gaming F15 FX506LH_FX506LH
- SSD: Micron_2210_MTFDHBA512QFD

4.1 Arquitectura del sistema

En esta parte se define los diferentes componentes de los cuales está compuesto el proyecto, además de las diferentes fases preestablecidas con sus respectivas tecnologías. Entre los procesos que se han definido en la solución propuesta y las herramientas que se han elegido para llevar a cabo el proyecto se ha diseñado esta arquitectura:

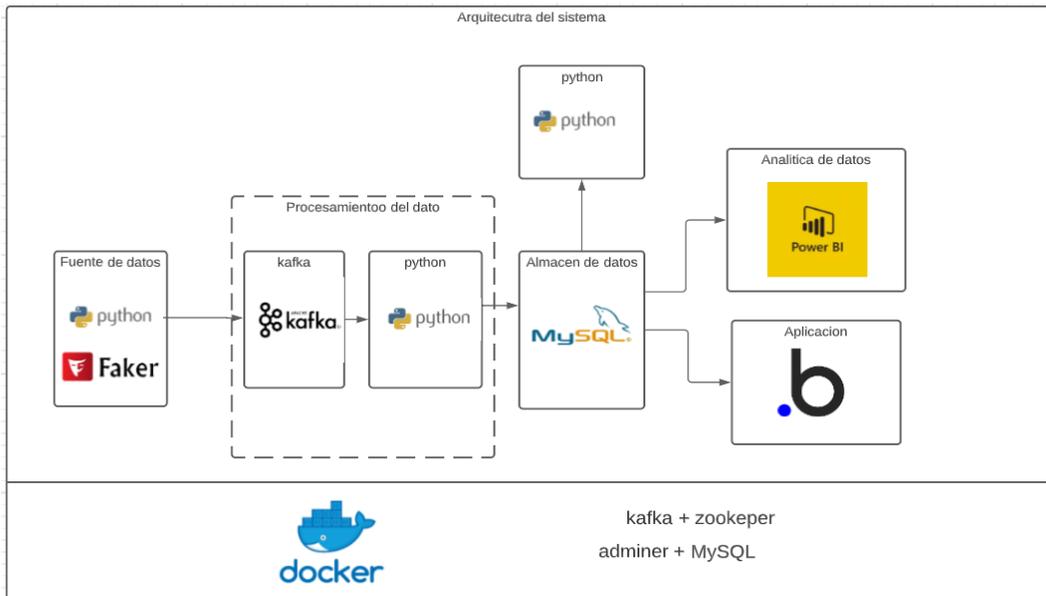


Ilustración 3 -Arquitectura del sistema

Lógicamente, antes de empezar a movilizar los datos entre los procesos, lo primero es lanzar en Docker las tecnologías que se puedan lanzar en contenedores mediante sus imágenes, específicamente se van a lanzar las correspondientes a las MySQL y a las de Kafka. Aunque solamente sean dos tecnologías como tal, es necesario lanzar un mayor número de contenedores ya que para el correcto funcionamiento de estas cada una necesita de otro software para poder gestionarlas. Para MySQL, aparte del mismo software de MySQL, se necesita un software llamado adminer, que es el que permite gestionar el contenido de la base de datos, y, por tanto, poder aplicar modificaciones sobre ellas. Y por la parte de Kafka, esta el software que lanz nos encontramos con zookeeper que es el software que gestiona el flujo de datos que van llegando al clúster de Kafka.

Ahora ya están disponibles las herramientas que se han lanzado en Docker, entonces, ya se puede empezar a desarrollar el resto de la arquitectura. La fuente de datos es la parte donde se obtienen los datos con los que se va a trabajar, ya se ha comentado en el punto anterior que no se disponen como ellos como tal por tanto era necesario realizar una simulación de los mismo. Por eso se ha decidido generarlos automáticamente mediante código. El lenguaje de programación utilizado ha sido python apoyados por una librería llamada *Faker*, esta librería aporta valores realistas de manera aleatoria. Con esto se es capaz de crear datos aleatorios que ayudaran a realizar una simulación de datos reales. Con la generación de los datos realizada, llega la parte de gestión del dato o ingeniería del dato, que corresponderían a los dos siguientes procesos: el de tratamiento del dato y el del almacenamiento

El procesamiento del dato tiene dos partes, una que se dedica a la ingesta de los datos y a la gestión del flujo de mensajes, por otro lado, la transformación de los datos para estructurarlos de la forma deseada y obtener datos nuevos a partir de los que llegan, para posteriormente volcarlos en la base de datos.

En la base de datos que es donde se almacena toda la información acerca de los clientes, para eso con anterioridad a realizar la carga en el paso anterior se debe haber creado la base de datos en el sistema de MySQL, describiendo la estructura que tendrá y definiendo cada una de las tablas que la componen con sus variables. Las dos tecnologías que se utilizan para implementar los siguientes procesos deberán conectarse con la base de datos para ir consultando la información.

También se utiliza otro script que se encarga de contabilizar quien cumple cada uno de los retos que se han propuesto y contabilizar la cantidad de retos que han sido superados, a su vez también calcular la puntuación que van acumulando por el cumplimiento de los retos. Una vez hechos los cálculos estos datos se cargarán en la base de datos. Con esto ya termina la parte dedica a la gestión del dato. Ahora ya queda elaborar los dos servicios que se ofrecen

Empezando por la parte más dedicada a la empresa (B2B), está el estudio de los clientes, donde se hace un análisis descriptivo con la herramienta de PowerBi. Se hacen distintos informes reflejados en un dashboard, que, mediante gráficas, se realiza un seguimiento de los usuarios y muestra su estado de salud y su desarrollo.

En el lado del producto para el cliente (B2C), se desarrolla la aplicación base de datos. Se crea una interfaz con donde cada usuario puede ver su progreso y los retos que van apareciendo, y se realiza una conexión con la base de datos desde Bubble para que la pagina se vaya actualizando conforme a los datos

4.2 Diseño detallado

Se ha visto cuales serían los procedimientos y pasos para realizar que se han diseñado para nuestro sistema, y contado con brevedad que es exactamente la tarea que se desempeña en cada una de las partes por las que está compuesta la arquitectura. Aquí, se desea explicar más detalladamente alguno de los componentes y partes por la que está compuesta para así entender más que es lo que se va a realizar y de qué manera.

Datos iniciales

Para la creación de simulación de los datos, ha habido que investigar acerca de información personal relacionada con la salud, y después información que recogerían dispositivos que miden este tipo de actividades. Se ha intentado aproximarse lo máximo posible a un modelo real, y generar datos que realmente obtendría una empresa en caso de aplicar este modelo.

Variable		Tipo	Descripción
id		Varchar(20)	Valor que identifica a cada uno de los clientes
name		Varchar(20)	Nombre del cliente
last_name		Varchar(20)	Apellido del cliente
position	lat	float	Posición en la que se va encontrando el cliente dividido en latitud y longitud
	lon	float	
age		int	Edad del cliente
gender		Varchar(20)	Genero del cliente
weight		float	Peso del cliente
height		float	Altura del cliente
cholesterol		float	Nivel de colesterol del cliente
drink		int	Indicador que marca si el cliente consume alcohol o no
smoker		int	Indicador que marca si el cliente fuma o no
disability		int	Indicador de si el cliente tiene discapacidades
heartbeats		float	Ritmo cardiaco que va teniendo el cliente
Previouspatology		int	Indicador de si anteriormente ha tenido algún tipo de patología

Como se ve en esta tabla aparece la información personal acerca de los usuarios, y luego ya vemos que aparece variables que están relacionadas con mediciones que se van realizando con relación a la actividad física del momento, como puede ser el ritmo cardiaco o la posición. Utilizamos la librería *faker* para simular algunos de los datos, y luego otros generados también mediante librerías que otorga valores aleatorios entre rangos numéricos.

Modelo relacional de la base de datos

Para el modelado de la base de datos ha habido que pensar de que información disponemos y que información necesitamos al final. Teniendo eso en mente se ha diseñado esta estructura la cual será la que hay que basarse cuando se crea la base de datos.

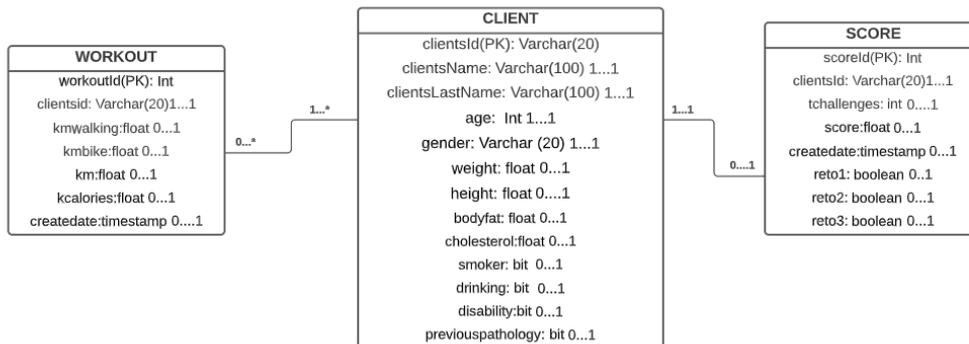


Ilustración 4 - Diseño relacional de la base de datos

En este modelo se puede ver que la información se ha separado en 3 tablas diferentes. La tabla del medio, la cual se llama *clients*. En esta tabla será se almacena la información personal del cliente relacionado con la salud, como puede ser su nombre, edad, altura, ...etc. La tabla *workout* es la que se va recopilando el ejercicio físico que se realiza a diario cada cliente, por eso, las claves primarias se le ha asignado id del cliente y a la fecha ya que no puede haber dos valores del mismo día que coincidan, además de conectar el id del cliente de la tabla workout con la tabla clients, para confirmar que los valores que se registran en la tabla workout es un usuario cliente registrado. Por último, está la tabla de score, aquí es donde se almacena toda la información con relacionada con el tema de los beneficios, es decir, que retos se han cumplidos, cuantos lleva, los puntos que va acumulando, y que retos han sido cumplidos y cuales no. Al igual que en la tabla anterior, también se conecta con la tabla del cliente mediante el id.

Consumidores y producer

Para entender exactamente como funciona los servidores de Kafka, simplemente una pequeña explicación de como conectar las otras herramientas con Kafka. Como se puede observar en la imagen de abajo, vemos como hay dos partes el consumer y el producer, estos son API que proporciona Kafka para que los distintos lenguajes permitan comunicarse con el clúster de Kafka.

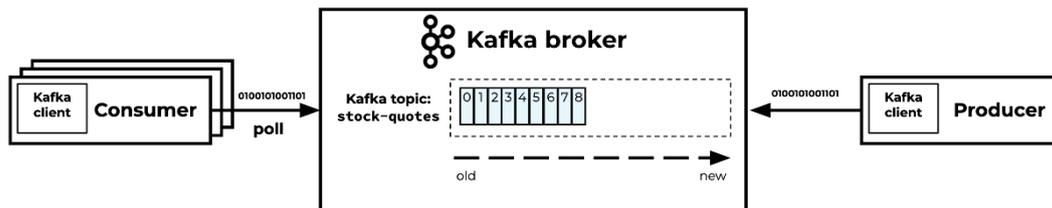


Ilustración 5- Sistema de comunicación de kafka

El consumer es el encargado de enviar los datos al clúster de Kafka bróker, la manera que tiene de enviarlos es primero mandándole la información al clúster del topics a los que se quiere conectar donde mandara la información que se va generando, y por otro lado tenemos el producer se conecta de la misma manera con el clúster, pero indicándole que es subscriptor y por tanto que a el se le deben enviar los mensajes que lleguen a los topics a los cual ha pedido suscribirse. La imagen de arriba corresponde con ese proceso el cual se le manda la información al cluster de donde quieren conectarse.

Métricas

Para los informes se establece una serie de métricas a representar mediante graficas para representar información importante acerca de los clientes y del proyecto. Es importante antes de empezar a realizar un informe, establecer que información es la que se quiere mostrar, viendo que es lo que te puede aportar esa información. Por ello se establecen que métricas se van a representar en las gráficas en cada uno de los informes. Se han separado las métricas en relación con cuál se usa en cada uno de los dashboard, en un dashboard se mostrará información personal acerca de los clientes de la aseguradora, y por otro lado se estudiará la evolución de los clientes en el aspecto físico. A continuación, se muestra cuales son las métricas representadas en cada uno de los informes

Dashboard-1

Métrica	Objetivo
Porcentaje de fumadores	Estudiar la salud de los clientes
Porcentaje de bebedores	Estudiar la salud de los clientes
Porcentaje de gente con discapacidad	Observar si tenemos una amplia clientela para realizar más ayuda a estos clientes
Contabilizar cuantos clientes ahí por etapas de vida	Saber que tipo de clientes tiene la empresa en función de su edad
Contabilizar cuantos clientes ahí por riesgo de obesidad	Evaluar el grado de riesgo de obesidad de los clientes

Dashboard-2

Métrica	Objetivo
Contabilizar los kilómetros totales en el tiempo	Comprobar si los clientes realizan más actividad física cada vez
Contabilizar los kilómetros recorridos en bici en el tiempo	Saber si los clientes utilizan mucho la bici para posibles futuras promociones
Contabilizar los kilómetros recorridos a pie en el tiempo	Saber si los clientes realizan muchas tareas de footing/runner para futuras promociones
Contabilizar cuanto se ha cumplido cada reto	Analizar cuáles son los retos que más gustan, o que resulten más interesantes

4.3 Tecnología utilizada

- **Docker**

Docker es un proyecto de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software. Se ha escogido utilizar Docker ya que otorga una gran cantidad de beneficios y es la aplicación por excelencia para el uso de contenedores. Hoy en día muchas empresas lo están aplicando a sus negocios



Ilustración 6-Logo de Docker

- **Kafka**



Ilustración 7 - Logo de kafka

Apache Kafka es una plataforma de transmisión de eventos distribuidos de código abierto utilizada por miles de empresas para canalizaciones de datos de alto rendimiento, análisis de transmisión, integración de datos y aplicaciones de misión crítica. Es un sistema horizontalmente escalable y tolerante a fallos. Permitirte publicar y suscribirte a flujo de datos (similar a una cola de mensajes), contar con un sistema de almacenamiento de registros persistente, y procesar flujos de datos en tiempo real.

- **Python**

Python es un lenguaje de programación de alto nivel que se utiliza para desarrollar aplicaciones de todo tipo. A diferencia de otros lenguajes como Java o .NET, se trata de un lenguaje interpretado, es decir, que no es necesario compilarlo para ejecutar las aplicaciones escritas en Python, sino que se ejecutan directamente mediante un interprete



Ilustración 8- Logo de python

Este lenguaje es el que va a ser usado en la mayor parte del proyecto, especialmente en la parte de Kafka. Se ha decidió utilizar Python por varias razones, una de las razones es por la comodidad al utilizar la herramienta, otra es porque es uno de los lenguajes más intuitivos y fáciles de comprender, a diferencia de otros lenguajes como poder se java o scala, y por último porque actualmente se encuentra en auge, siendo de los lenguajes más utilizados y continúa creciendo.

- **MySQL**



Ilustración 9 - Logo de MySQL

MySQL es un sistema de gestión de bases de datos relacionales de código abierto respaldado por Oracle y basado en el lenguaje de consulta estructurado (SQL). Mysql es la base de datos mas conocida actualmente, proporciona un alto rendimiento, debido a que permite configurar la base de datos para una gran variedad de operaciones, además de ser un sistema escalable y flexible, y ofrece uno de los motores de datos transaccionales más potentes del mundo.

- **PowerBI**



Ilustración 10 - Logo de PowerBI

Power BI es un servicio de análisis de datos de Microsoft orientado a proporcionar visualizaciones interactivas y capacidades de inteligencia empresarial con una interfaz lo suficientemente simple como para que los usuarios finales puedan crear por sí mismos sus propios informes y paneles. Las herramientas de BI más destacables para realizar análisis son Power BI y Tableau. Entre ellas dos se ha decidido trabajar con power Bi ya que resulta más familiar y además

- **Bubble**

Bubble es un lenguaje de programación visual, una plataforma de desarrollo low-code y una plataforma de aplicaciones como servicio, desarrollada por Bubble Group, que permite a personas crear aplicaciones web. Es una herramienta muy funcional, ya que dispone de una gran cantidad de plugins que permite que podamos realizar una gran variedad de cosas. También facilita API para poder conectar con otras herramientas, y mas a tener en cuenta, debido al proyecto, con bases de datos de MySQL. Además, es una plataforma muy intuitiva y amigable



Ilustración 11- Imagen de bubble

4. Desarrollo de la solución propuesta

En el desarrollo de la solución se trata paso a paso cada uno de los procesos definidos en la arquitectura. Se explican los distintos puntos como se han realizado cada uno de los procesos que componen y como se consigue conectar las distintas tecnologías para funcionen los procesos

Como se menciona en la arquitectura, se empieza con el despliegue de las herramientas en docker. En ella lo que se debe hacer es crear los contenedores necesarios para las distintas tecnologías que componen la arquitectura del sistema desde el lanzamiento de los datos hasta la base de datos de mysql, mediante sus imágenes.

Para el despliegue se ha creado un archivo llamado docker-compose. En este archivo se encuentra toda la información para que Docker realice el despliegue, se explica los contenedores a usar, que imagen tiene que usar cada contenedor, los puertos a los que se debe conectarse y más información que se le puede aplicar a los contenedores. Es mejor realizando de esta manera, en vez de realizar el lanzamiento uno a uno, ya que permite lanzar todos los contenedores de una vez, pudiendo establecer un orden de lanzamiento si fuese necesario, y además de poder concentrar la configuración de todos los contenedores en un archivo. Como se ha dicho anteriormente las tecnologías a desplegar en docker son las correspondientes a los softwares de MySQL y Kafka y los programas necesarios para gestionarlos

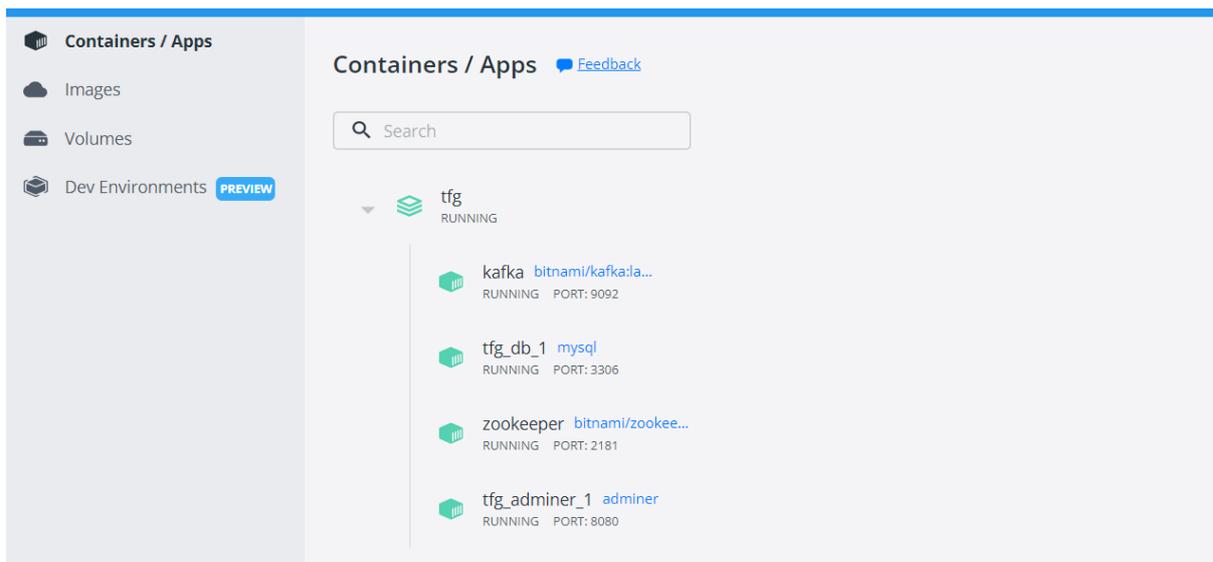


Ilustración 12 - Contenedores lanzados en docker

4.1 Fuente de datos

Con las herramientas disponible, se empiezan a crear los datos para elaborar la solución propuesta. Se realiza un script donde se crean y se otorgan valores a las variables de inicio. En aquellas variables numéricas o booleanas se han podido utilizar la librería `random` para que aleatoriamente asignase variables dentro de un rango. En cambio, en aquellas nominales como puede ser el nombre o crear un id, ha habido que recurrir se ha utilizado la librería `faker` para que nos otorgase datos suficientemente variados para que el estudio funcionase correctamente. Para realizar actualizaciones se ha creado una función la cual va actualiza los valores de posición de cada uno de los clientes que hemos creado cada vez que se le llama, cuando se vayan a enviar los datos se hará una llamada

```
def generar_datos():
    global users
    for i in range(0,USERS_TOTAL):
        user={}
        user["id"]=faker.ssn()
        user["name"]=faker.first_name()
        user["last_name"]=faker.last_name()
        user["position"]={"lat":random.uniform(39.4, 36.0144551),"lon":random.uniform(-0.3, -0.4)}
        user["age"]=random.uniform(16, 85)
        user["gender"]=random.choice(["man","woman"])
        user["weight"]=random.uniform(60, 110)
        user["height"]=random.uniform(150, 210)
        user["bodyfat"]=random.uniform(3, 45)
        user["cholesterol"]=random.uniform(150, 300)
        user["smoker"]=random.choice([0,1])
        user["drinking"]=random.choice(["0","1"])
        user["disability"]=random.choice(["0","1"])
        user["heartbeats"]= random.uniform(60,200)
        user["previouspatology"]=random.choice(["0","1"])
        user["time"]=datetime.now().strftime("%Y-%m-%d %H:%M:%S.%f")
        users[user["id"]]=user
    num=0

    print("Datos generados")

def actualizar_posicion():
    global users
    global old
    old = copy.deepcopy(users)
    if len(users)>0:
        print("STEP")
        for element in users.items():
            users[element[0]]["time"]=datetime.now().strftime("%Y-%m-%d %H:%M:%S.%f")
            lat=users[element[0]]["position"]["lat"]
            lon=users[element[0]]["position"]["lon"]
            users[element[0]]["position"]["lon"]=lon+random.uniform(0, 0.00003)
            users[element[0]]["position"]["lat"]=lat+random.uniform(0, 0.00003)
```

Ilustración 13-Obtención de los datos de imágenes

4.2 Base de datos

Antes de empezar con el procesamiento de los datos, se crea la base de datos que se va a utilizar. Esto se hace al principio para así cuando se lancen los datos hacia la base de datos ya se encuentren disponibles para cargar en ella los datos.

Para acceder al sistema de MySQL que se ha lanzado, hay que conectarse mediante el puerto:8080 del propio equipo, que es que ha sido asociado desde el contenedor de docker, poniendo desde cualquier navegador la dirección *localhost:8080*. Cuando se accede aparece la página de registro donde hay que registrarse con usuario y contraseña. Para poder realizar modificaciones en el proyecto se accede como administrador. En nuestro caso accederemos con los que vienen por defecto, que nos garantiza todos los permisos sobre el sistema. Una vez dentro aparece la página inicial de gestión de MySQL. Estando dentro ya se crea la base de datos tal cual, y luego una vez creada la base de datos, ya se pueden empezar a crear su estructura y definir reglas sobre ella

Idioma: Español MySQL » db

Adminer 4.8.1 **Seleccionar Base de datos**

DB:

[Comando SQL](#) [Importar](#) [Exportar](#)

[Crear Base de datos](#) [Privilegios](#) [Lista de procesos](#) [Variables](#) [Estado](#)

Versión MySQL: **8.0.29** a través de la extensión de PHP **MySQLi**

Logueado como: **root@172.22.0.3**

	Base de datos - Refrescar	Colación	Tablas	Size - Compute
<input type="checkbox"/>	besecure	utf8mb4_0900_ai_ci	?	?
<input type="checkbox"/>	information_schema	utf8_general_ci	?	?
<input type="checkbox"/>	mysql	utf8mb4_0900_ai_ci	?	?
<input type="checkbox"/>	performance_schema	utf8mb4_0900_ai_ci	?	?
<input type="checkbox"/>	sys	utf8mb4_0900_ai_ci	?	?

Selected (0)

Ilustración 14-Página inicial de MySQL

Una vez dentro se definen mediante comando SQL cada una de las tres tablas. Como tanto la tabla de score como la de workout esta enlazada a la tabla clients, mediante client id se debe crear una restricción de clave ajena dentro de ambas tablas, además se ha añadido que la eliminación sea en cascada, así cuando se elimine algún cliente de la tabla clients, se eliminarán los registros de las otras tablas que estén relacionados con estas.



Ilustración 15-Estructura de la base de datos

Como se observa, la estructura es idéntica al modelo de datos que se ha definido. Por lo tanto se sabe que la estructura se ha construido como se había establecido en un principio. Con la base de datos ya construida ya se realizan las tareas de procesamiento de datos

4.3 Procesamiento de datos

El mismo script donde se han definido los datos, va a actuar como Productor de Kafka. Mediante las librerías de Kafka se crea un objeto que actúe como productor indicándole donde se encuentran los servidores de Kafka y a que topics debe mandarlos. Este objeto es al que se le dan los datos para que mande los datos al servidor

```

def getKafkaProducer():
    producer = None
    while producer==None:
        try:
            producer = KafkaProducer(bootstrap_servers=['localhost:9092'],value_serializer=lambda x: dumps(x).encode('utf-8'))
  
```

Ilustración 16- Creación del producer en python

Los datos ya se encuentran dentro del clúster de Kafka, ahora hay que crear en el script donde se realizaran las modificaciones un objeto que actúe como consumer. Al igual que en el caso anterior se debe indicar donde se encuentran los datos, tanto el cluster de Kafka como los topics.

```

consumer = KafkaConsumer(
    'mocked_data',
    bootstrap_servers=['localhost:9092'],
    auto_offset_reset='earliest',
    enable_auto_commit=True,
    group_id='my-group-id',
    value_deserializer=lambda x: loads(x.decode('utf-8'))
)
  
```

Ilustración 17- Creación del consumer en python

Con los datos recibidos desde kafka, ya se pueden empezar a realizar modificaciones y obtener nueva información de los datos antes de mandarlos a la base de datos que ha sido creada anteriormente.

Una de las informaciones que se pueden obtener de los datos iniciales es los kilómetros que se recorren. Para calcular la distancia hemos necesitado recurrir a los valores de posición que teníamos anteriormente y a partir de ahí con la posición actual se puede realizar un cálculo de la distancia recorrida mediante la formula del semiverseno

$R = \text{radio de la Tierra}$

$\Delta\text{lat} = \text{lat2} - \text{lat1}$

$\Delta\text{long} = \text{long2} - \text{long1}$

$a = \sin^2(\Delta\text{lat}/2) + \cos(\text{lat1}) \cdot \cos(\text{lat2}) \cdot \sin^2(\Delta\text{long}/2)$

$c = 2 \cdot \text{atan2}(\sqrt{a}, \sqrt{1-a})$

$d = R \cdot c$

Ilustración 18 – Fórmula de semiverseno

También se necesita calcular las calorías que se van perdiendo, para saber la cantidad exacta que se pierdo se ha utilizado la distancia recorrida con la función anterior, el ritmo cardiaco y también utilizando la tabla que aparece abajo la cual describe la cantidad de calorías que se quema en 30 minutos, para el proyecto se ha adaptado al intervalo de tiempo que hay entre la recepción de los datos. También se utiliza la distancia recorrida en ese periodo para averiguar si el medio que esta utilizando es a pie, en bici o en un vehículo.

Correr	325
Andar sin prisas (1,5-3 km/hora)	60-75
Andar sin prisas (4,8 km/hora)	150
Andar más rápido (6 km/hora)	180
Andar muy rápido (7,5 km/hora)	200-240

Ilustración 19- Calorias quemadas segun el ejercicio

Con todos los datos disponibles, se establece una conexión con la base de datos para poder enviar los datos, como los datos están separados en tablas, hay que hacer distintas consultas y llamadas a la base de datos indicar a donde queremos mandar los datos además de indicar que datos son los que deseamos mandar.

Para la tabla de clients como no es necesario actualizar datos, simplemente se hace una query para insertar los datos en la tabla.

Luego, para la inserción de datos y actualización de la columna de workout ha creado un procedimiento en la base de datos que ejecutara cada vez que desde el script se lance una llamada a dicho procedimiento. En el procedimiento se define que si detecta que el cliente no contiene un registro en ese día en la tabla de workout se creará un registro con el id del cliente poniendo a 0 los distintos valores, si encuentra un registro en la tabla lo que hará es basarse en la fecha y en el id modificar el registro correspondiente y modificar los valores sumando los datos que se le envían a los que ya se encuentran en la base de datos.

Modificar procedimiento: InsertUpdateKM

Nombre:

	Nombre de Parámetro	Tipo	Longitud	Opciones	+
IN	id	varchar	20	(colación)	+ ↑ ↓ ✕
IN	kmxwalking	float			+ ↑ ↓ ✕
IN	kmxbike	float			+ ↑ ↓ ✕
IN	kxcalories	float			+ ↑ ↓ ✕
IN	date	datetime			+ ↑ ↓ ✕

```

BEGIN
  IF (select count(*) from workout where clientsId=id and createDate = date_format(date, "%Y-%m-%d"))>0
  ELSE
  INSERT INTO besecure.workout (clientsId, kmwalking, kmBike, km, kcalories, createDate) VALUES (id, 0,
  END IF;
END
  
```

Ilustración 20- Procedimiento para actualizar tabla workout

Solo faltan los datos de la tabla score, como es necesario utilizar otro script para hacer los cálculos de si se cumplen los retos, solamente se insertarán en la tabla los id de cada cliente inicializando el resto de los datos a 0.

Con estos pasos ya hay un proceso para añadir la información de los clientes y para actualizar la tabla de workout, además de tener inicializado los valores de los clientes en la tabla score. Falta tener una manera de que los datos de la tabla score se vayan actualizando conforme se vayan cumpliendo los retos. Estos cálculos lo van a realizar el otro script donde se ira comprobando reto a reto si se van cumpliendo con cada uno de ellos. De momento los retos

que hay son los 3 siguientes con los requisitos de cada uno y la puntuación que se obtiene al cumplirlos.

Retos	Requisitos	Puntuación
Reto 1	Más de 100km en bici	50
Reto 2	Mas de 10 km a pie durante 5 días	100
Reto 3	Más de 400km en un solo día	80

Para cada uno de los retos se realiza una consulta a la base de datos donde se solicitan datos de la tabla score y de la tabla para poder comprobar primero si los retos ya se habían cumplido anteriormente y luego calcular si los ha cumplido. Si los cálculos están se hará una nueva consulta a la base de datos para actualizar los valores del cliente en dicha tabla. Habrá un temporizador para que cada cierto tiempo repita el proceso y compruebe con información actualizada si se cumplen con los retos. Con esto terminaría la parte de procesamiento y almacenamiento del dato.

4.4 Análisis de datos

Como se ha visto en la arquitectura utilizaremos la herramienta de Power Bi para realizar el análisis descriptivo acerca del proyecto.

Lo primero, lógicamente que se necesita es disponer de los datos para poder manejarlos, para ello hay que conectar la base de datos a la herramienta marcando donde se encuentra el servidor que almacena los datos y la tabla a acceder, aquí nosotros indicamos “localhost” ya que, desde local lanzado desde nuestro equipo, e indicando la base de datos que queremos utilizar para el análisis, que en esta ocasión será la de que hemos nombrado como *beseure* en la base de datos.

The image shows a dialog box titled "Base de datos SQL Server" with a close button (X) in the top right corner. It contains the following fields and options:

- Servidor**: A text input field containing "localhost".
- Base de datos (opcional)**: A text input field containing "beseure".
- Modo Conectividad de datos**: A dropdown menu with two options: "Importar" (selected with a yellow dot) and "DirectQuery".
- Opciones avanzadas**: A link with a right-pointing chevron (>).
- Buttons**: Two buttons at the bottom right: "Aceptar" (highlighted in yellow) and "Cancelar".

Ilustración 21-Conectar PowerBi con la base de datos

Los informes son representados mediante los dos dashboard que se construyen. El primero que se realiza es el relacionado con la salud de los clientes de la empresa. Para realización de este informe ha habido que realizar algunos cálculos para obtener la información. Para clasificar a los usuarios en base a su masa corporal ha sido necesario investigar acerca de los niveles de grasa del cuerpo que hay que tener para clasificar a cada uno de los grupos. Con ello y la variable "bodyFat" de la tabla cliente se ha podido realizar la clasificación



Ilustración 22-Indicador de obesidad

Estando creadas las medidas necesarias en el PowerBi, ya pueden construir cada una de las gráficas que representan las métricas que se habían preestablecido en el diseño y poder diseñar el informe. En la imagen de abajo se muestra el dashboard completo que se ha construido y que conformaría lo que sería el informe

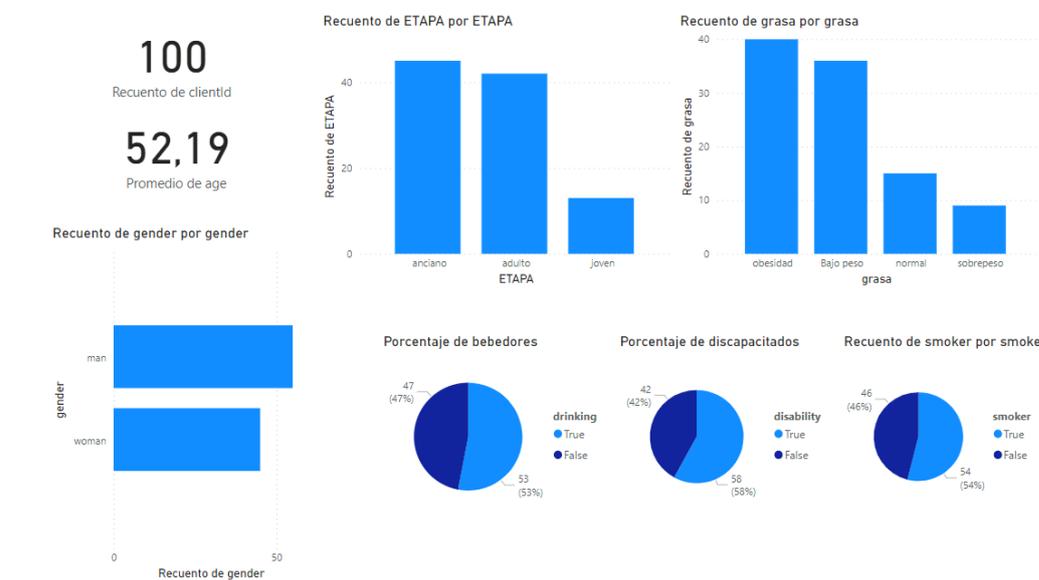


Ilustración 23-Dashboard 1

Para el segundo informe se utilizan tanto la tabla de workout como la tabla de score. A diferencia del primero, no es necesario realizar ningún cálculo excepcional, los únicos cálculos a realizar será el sumatorios de los kilómetros representados en la tabla workout para obtener la totalidad de un dato. para representar las métricas que se han propuesto para este informe, ya que se dispone de la información dentro de la base de datos. Así es como se ha diseñado el segundo informe partiendo de las gráficas predefinidas.

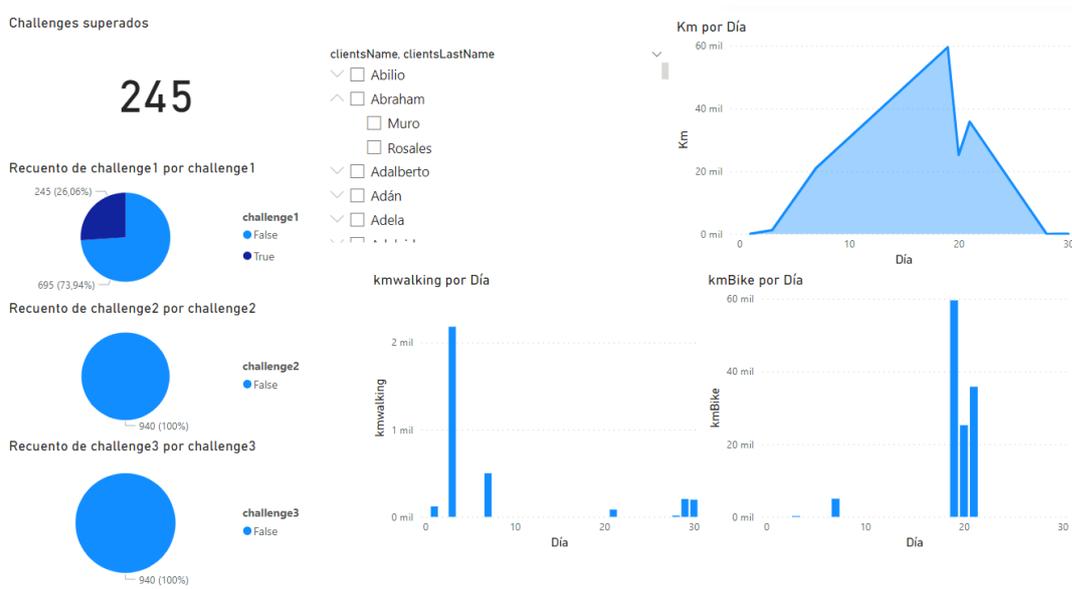


Ilustración 24 - Dashboard 2

Aplicación de beneficios

Ahora es el turno de diseñar y crear la aplicación que finalmente utilizarían los clientes para poder ver su rendimiento y desarrollo conforme al desempeño que van realizando. Como ya se ha mencionado utilizaremos la aplicación de Bubble para su desarrollo.

Para empezar antes de realizar cualquier tipo de diseño de la aplicación vamos a conectar con la base de datos para que desde Bubble se puedan obtener los datos que se van generando. Para eso, la aplicación dispone de una serie de pluggins para instalar en la aplicación que permite instalar una API para poder conectar con base de datos que utilizan los estándares de bases de datos de SQL, como puede ser MySQL o PostgreSQL. Una vez estén instalados hay que indicar correctamente las credenciales que te indica la aplicación para que pueda conectarse.

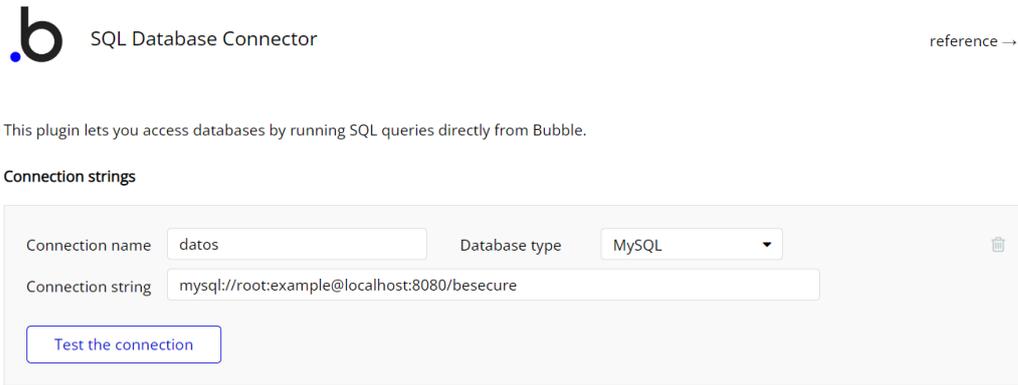


Ilustración 25- Conexión a la base de datos

Para utilizar los datos que se quiere mostrar en la interfaz de la aplicación, es necesario además hacer una query donde indicamos cuales son los datos de la base de datos que queremos obtener. Para realizar las consultas desde el programa hay que escribirlo en lenguaje SQL, ya que facilita este uso debido a que este tipo de bases de datos están basados en dicho lenguaje. Por tanto, se escribe la consulta de la misma manera que se realizaría en la base de datos para obtener la información.

Se realizan dos queries una con los datos que aparecerán de diario y otra con el progreso general. Para la primera simplemente se escogen los datos que se encuentran en la tabla workout correspondientes al día actual, ya que se muestran para cada cliente la mayoría de los atributos de esta tabla. Y luego por otro lado se utiliza tanto la tabla workout para hacer recuento de la información general y además mostrar la información de la puntuación

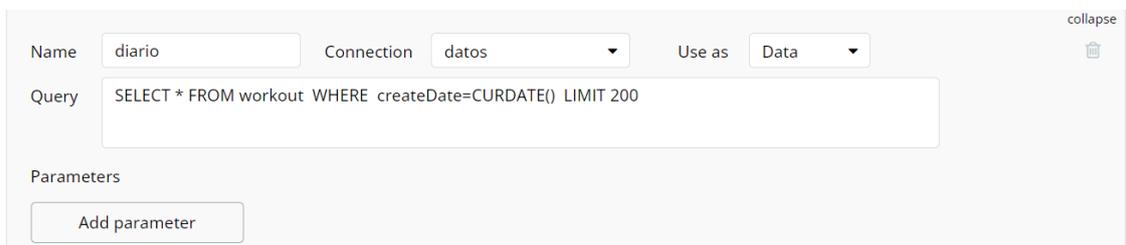


Ilustración 26 - Query acerca de la información diaria

Ahora ya se encuentran los datos cargados en Bubble y por tanto se podrán mostrar en la aplicación. A partir de aquí se crea la interfaz de usuario. Se podría decir que la interfaz la hemos dividido en 3 partes diferentes.

En la primera parte donde se enseñará el desempeño que lleva en ese mismo día, luego por otro lado tendríamos la parte donde se mostraría el progreso que va teniendo, donde se ve el score acumulado la cantidad de retos cumplidos, etc.... Y ya por último aparecen los distintos retos que pueden ir realizando los clientes, y también la puntuación que se le otorgaría a los clientes en caso de cumplimentar dichos retos. Con esto ya tenemos terminada la aplicación, y por tanto, la última parte del proyecto.

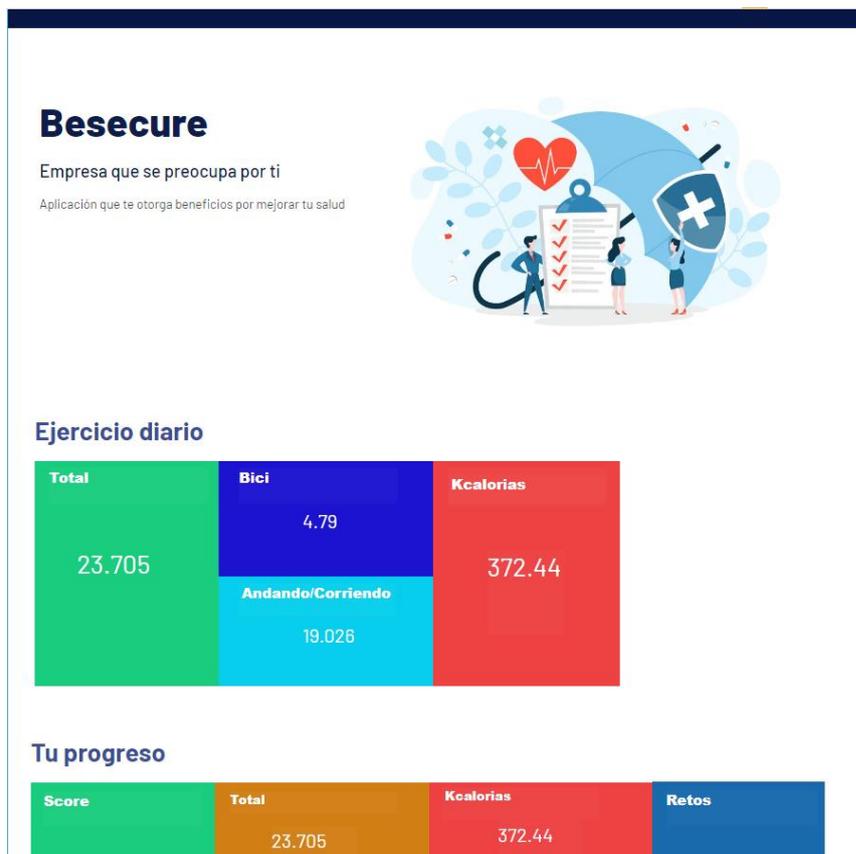


Ilustración 27 - Pagina de la aplicación

5. Implantación

En este apartado se presenta la etapa de implantación de la solución. En el desarrollo de la solución hemos visto cómo se han ido desarrollando cada una de las tareas a realizar para que funcionasen cada uno de los procesos y en el orden en que deben ser realizados para que funcionasen correctamente. Ahora en la parte de implantación se verá la puesta en producción del sistema, para enseñar como se realiza a continuación se dejará el enlace a un video donde lo mostrará. En el video veremos una demo de como funciona lanzando los distintos scripts y desplegando las aplicaciones. Hay algunas que ya estarán lanzadas como pueden ser las bases de datos, o los productos, ya que han sido creadas durante el desarrollo del sistema y ya se encuentran realizadas. En el video simplemente veremos cómo van moviéndose los datos desde el inicio hasta llegar a los distintos componentes y como afectan a estos.

Video de implantación del proyecto: https://youtu.be/oY-zs9RQ2_U

6. Conclusiones

Al final, el objetivo principal que se ha conseguido construir los dos productos que estaban planteados. La realización de un seguimiento de la evolución de los clientes, con respecto a la actividad física, como el desarrollo de una aplicación para obtener beneficios. Se ha conseguido realizar informes que muestran el desarrollo de los clientes y también desarrollar la aplicación donde se puede observar el proceso de los clientes

Otro objetivo principal importante era mostrar la utilidad que tiene este tipo de herramientas, con este sistema, después del desarrollo donde se ve su funcionamiento y como se ve más adelante el cómo se puede aplicar a distintos campos, ha habido un estudio de las tecnologías y se ha descubierto nuevas técnicas y métodos a lo largo del proyecto.

Esto se ha conseguido en gran parte por el cumplimiento de los objetivos específicos que se habían establecidos para conseguir este objetivo principal, vamos a ir tratando uno por uno, haber en qué medida se han logrado y como se han tratado:

- Para la arquitectura física para tener los datos almacenados y normalizados, se ha creado la base de datos la cual se ha visto que almacena toda la información correctamente, además de manera estructurada.
- Para la interfaz de usuario se ha creado mediante Bubble una interfaz sencilla que es de fácil entendimiento, y en cuanto a funcionalidad se podría haber añadido alguna función más para poder interactuar con la aplicación.
- En cuanto a los datos en tiempo real, se ha conseguido gestionar la información mediante la tecnología de Kafka, para que los datos lleguen en su totalidad a la base de datos.
- Realización de informes que represente información útil. A falta de una empresa que realmente podría decir que información resultaría, para establecer las métricas, se ha investigado y ha observado con que objetivo se ha decidido cada métrica para ver la importancia de esta

Se considera que la mayoría de los objetivos han sido cumplidos correctamente, pero es verdad que no se ha llegado a alcanzar el grado de perfección que nos gustaría haber conseguido, por algunos de los puntos tratados anteriormente. Pero en general, el grado de

satisfacción es alto ya que nos enfrentábamos a un proyecto al cual se había tocado de forma muy leve estas tecnologías. Se ha aprendido a elaborar metodologías para proyectos de este tipo de índole, además de indagar más en los procesos que este comprende. Lógicamente también se ha aprendido bastante teoría acerca de la gestión de los datos, de los procesos y de los componentes que están implicados

7. Trabajos futuros

A la hora de hablar de posibles trabajos a realizar en un futuro, se va a hablar desde dos perspectivas diferentes; por un lado, la parte de cambios y modificaciones que se podrían hacer a este proyecto para mejorar su eficiencia y su eficacia, y así obtener un mejor producto. Y por otro lado que otros proyectos se podrían crear a partir de las tecnologías utilizadas y los conocimientos adquiridos; y que se han ido viendo a lo largo de este proyecto.

Primero vamos a tratar la parte que correspondería a este mismo sistema. Las posibles mejoras y modificaciones que se han contemplado son las siguientes:

- Alguna de las posibles mejoras a poder aplicar en este proyecto es la escalabilidad del mismo. Con esto se hace referencia a la parte de llegada de mensajes de que el sistema pueda escalar en correspondencia a la cantidad de mensajes que se reciben, para ello se puede utilizar kafka stream o apache spark que ayudan a la escalabilidad repartiendo la carga de trabajo.
- También se podría mejorar la parte de la aplicación, haciendo que la aplicación disponga de más funciones, haciéndola más funcional y realizando diseño que resulte aún más amigable.
- Trabajar con datos veraces. Como ya se ha visto los datos que han sido utilizados han sido una demo de como seria en un sistema real. Sería interesante probar este sistema con datos de clientes y datos recogidos reales.

Hablando de otros campos donde poder adquirir sistemas parecidos al que se ha visto en el proyecto o aplicar conocimiento que se han visto a lo largo de él, tenemos muchos campos distintos donde se podrían aplicar. Ya se ha mencionado anteriormente, pero cada vez mas a las empresas les interesa el mundo del Big Data y de la gestión de los datos viendo los resultados que se pueden obtener de estos. Existen muchos ejemplos como puede ser para el deporte el mostrar como esta yendo en un partido mostrándole a los clientes las estadísticas de los partidos en vivo, o simplemente sin la necesidad de tener datos continuamente en la recepción, coger datos anteriores de una empresa y una vez analizados, observar las oportunidades que existen para crear una aplicación para fomentar dicha idea. Estos son unos simples ejemplos, como se puede ver existen muchas maneras diferentes de aplicar este tipo de sistemas a diferentes campos, por eso se considera de gran utilidad este tipo de tecnologías y su aplicación al negocio, por lo que puede llegar a aportar a un negocio.

8. Glosario

- **Business intelligence.** Hace referencia el conjunto de estrategias, aplicaciones, datos, productos, tecnologías y arquitectura técnicas, los cuales están enfocados a la administración y creación de conocimiento sobre el medio, a través del análisis de los datos existentes en una organización o empresa.
- **Dashboard.** Un dashboard es una herramienta de gestión de la información que monitoriza, analiza y muestra de manera visual los indicadores clave de desempeño (KPI), métricas y datos fundamentales para hacer un seguimiento del estado de una empresa, un departamento, una campaña o un proceso específico.
- **Ingeniería del dato.** La ingeniería de datos es el campo asociado con el análisis y las tareas para obtener y almacenar los datos de otras fuentes. Luego, procese esos datos y conviértalos en datos limpios que se utilizarán en otros procesos, como visualizaciones de datos, análisis de negocios, soluciones de ciencia de datos, etc.
- **API.** API significa “interfaz de programación de aplicaciones”. En el contexto de las API, la palabra aplicación se refiere a cualquier software con una función distinta. La interfaz puede considerarse como un contrato de servicio entre dos aplicaciones. Este contrato define cómo se comunican entre sí mediante solicitudes y respuestas.
- **Aislamiento(procesos).** Es un mecanismo de seguridad para separar los programas en ejecución, normalmente para mitigar los fallos del sistema y las vulnerabilidades del software para que no se propaguen.
- **Registro.** Es toda la información de un elemento que se almacena en un archivo o tabla de la base de datos
- **Clúster.** Grupos de servidores que se gestionan juntos y participan en la gestión de la carga de trabajo. Un clúster puede contener nodos o servidores de aplicaciones individuales. Un nodo suele ser un sistema físico con una dirección IP de host distinta que ejecuta uno o más servidores de aplicaciones.
- **Query.** Una query es una pregunta o consulta. En internet, en el contexto de acciones realizadas en motores de búsqueda, se refiere a cada consulta individual que se realiza en la web.
- **Procesamiento (datos).** Es la acumulación y manipulación de elementos de datos para producir información significativa

9. Anexo

OBJETIVOS DE DESARROLLO SOSTENIBLE

Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS).

Objetivos de Desarrollo Sostenibles	Alto	Medio	Bajo	No Procede
ODS 1. Fin de la pobreza.				
ODS 2. Hambre cero.				
ODS 3. Salud y bienestar.	x			
ODS 4. Educación de calidad.				x
ODS 5. Igualdad de género.				x
ODS 6. Agua limpia y saneamiento.				x
ODS 7. Energía asequible y no contaminante.				
ODS 8. Trabajo decente y crecimiento económico.				x
ODS 9. Industria, innovación e infraestructuras.		x		
ODS 10. Reducción de las desigualdades.				x
ODS 11. Ciudades y comunidades sostenibles.				x
ODS 12. Producción y consumo responsables.				x
ODS 13. Acción por el clima.		x		
ODS 14. Vida submarina.				x
ODS 15. Vida de ecosistemas terrestres.				x
ODS 16. Paz, justicia e instituciones sólidas.				x
ODS 17. Alianzas para lograr objetivos.				x

Reflexión sobre la relación del TFG/TFM con los ODS y con el/los ODS más relacionados.

- En alto grado se cumple con el objetivo de salud y bienestar, ya que es un trabajo en gran parte planteado para que la gente mejore su estado de salud, fomentando que

realicen ejercicios físicos. Esto provoca que disminuya el riesgo de sufrir lesiones o enfermedades, ya que queda demostrado que el deporte es una gran manera de disminuir el número de enfermedades

- También en su cierta medida está relacionada con la acción por el clima, ya que fomentar que la gente realice ejercicio, provoca que la gente no consuma los medios de transportes convencionales, como puede ser el coche, la moto etc..., que son contaminantes para el medio ambiente, y se acostumbre a ir a pie o en bicicleta que no contamina, y así mejorar el medio ambiente.
- Por último, también aporta industria, innovación e infraestructuras. El proyecto se ha construido una infraestructura para soportar los distintos procesos mediante las nuevas tecnologías que están surgiendo en la industria del Big Data recurriendo a métodos modernos, construyendo un sistema y un producto innovador.

10. Referencias

1. D. (2022, 20 abril). *Apache Kafka: escalabilidad y baja latencia en la gestión de datos en tiempo real*. Decide. <https://decidesoluciones.es/apache-kafka-gestion-de-datos-en-tiempo-real/>
2. Lafuente, A. (2018, 12 septiembre). *Bases de datos relacionales vs. no relacionales: ¿qué es mejor?* Aukera. <https://aukera.es/blog/bases-de-datos-relacionales-vs-no-relacionales/>
3. Senra, I. (2021, 27 julio). *Qué es una Query - Definición, significado y ejemplos*. Arimetrics. <https://www.arimetrics.com/glosario-digital/query#:~:text=Una%20query%20es%20una%20pregunta,datos%20o%20sistema%20de%20informaci%C3%B3n.>
4. Introducción: Clústeres. (2021, 25 junio). Documentación IBM. <https://www.ibm.com/docs/es/was-zos/9.0.5?topic=servers-introduction-clusters>
5. Coutinho, V. (2021, 12 febrero). KPIs: descubre qué son los indicadores clave de rendimiento y cómo usarlos para orientar tus estrategias. Rock Content - ES. <https://rockcontent.com/es/blog/kpis/>
6. Herrero, A. (2022, 28 enero). Ventajas y desventajas en una base de datos relacional. Escuela tecnológica especializada en programación, ciberseguridad, XR, IoT, IA y blockchain | CODE SPACE. <https://codespaceacademy.com/blog/ventajas-y-desventajas-base-de-datos-relacional/>
7. Miguel, J. (2011, 29 abril). ¿Cómo calcular la distancia entre dos puntos geográficos en C#? (Fórmula de Haversine). Genbeta. <https://www.genbeta.com/desarrollo/como-calcular-la-distancia-entre-dos-puntos-geograficos-en-c-formula-de-haversine>

8. Procesamiento de Datos: Definición, elementos y procesos. Tecnologías de información. <https://www.tecnologias-informacion.com/procesamientodatos.html>

9. Velázquez, A. (2021, 15 octubre). ¿Qué es el análisis descriptivo? QuestionPro. <https://www.questionpro.com/blog/es/analisis-descriptivo/#:%7E:text=El%20an%C3%A1lisis%20descriptivo%2C%20como%20su,y%20no%20tiene%20una%20hip%C3%B3tesis.>