INSTITUTO POLITÉCNICO NACIONAL

UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS

upiita·ipn

ACUSE REPORTE FINAL

INGENIERÍA TELEMÁTICA

El presente documento hace constar que se ha entregado el Reporte Final de **Trabajo Terminal I** (Trabajo Terminal I/Trabajo Terminal II) al Jurado para su defensa en la presentación **Extraordinaria** (Ordinaria/Extraordinaria/ETS), del proyectodenominando:

NÚMERO DE REGISTRO (Número asignado en la Constancia de Registro de Protocolo)	TTT-2022/2-19
TÍTULO	
SISTEMA PARA EL ANÁLISIS DE DATOS METEOROLÓGICOS Y DE CALIDAD DEL AIRE USANDO MICROSERVICIOS Y APRENDIZAJE AUTOMÁTICO	

Proyecto integrado por los siguientes alumnos:

	NOMBRE COMPLETO (Iniciando por apellidos y después nombres)
Alumno 1	APARICIO ESPINOZA OCTAVIO JOEL
Alumno 2	HERNÁNDEZ CASTILLO FRANCISCO
Alumno 3	LÓPEZ TIRADO NADIA JAQUELINE
Alumno 4	

A continuación firman de recibido todos los miembros del Jurado del proyecto:

DRA. MARTÍNEZ SEIS BELLA CITLALI 11878306 **PRESIDENTE** Nombre Completo del Presidente (Incluir el Grado Académico)	DR. CARLOS HERNÁNDEZ MEJÍA CP. 13030213 **SECRETARIO** Nombre Completo del Secretario (Incluir el Grado Académico)
DR. MIGUEL FÉLIX MATA RIVERA 4643102 **ASESOR 1** Nombre Completo del Asesor 1 (Incluir el Grado Académico) Interno [X] Externo [ ]	**ASESOR 2** Nombre Completo del Asesor 2 (Incluir el Grado Académico) Interno [ ] Externo [ ]
ASESOR 3 Nombre Completo del Asesor 3 (Incluir el Grado Académico) Interno [ ] Externo [ ]	M. EN C. NOÉ SIERRA ROMERO C.P. 6355794 **SUPLENTE** Nombre Completo del Suplente (Incluir el Grado Académico)

NOTAS IMPORTANTES:

- El Acuse de Reporte Final deberá entregarse en original al Secretario, este deberá incluir todas las firmas de los miembros del Jurado, incluyendo al Suplente, sin excepciones, en caso contrario el Departamento de Tecnologías Avanzadas no podrá programar la presentación del proyecto.
- Los alumnos tendrán la responsabilidad de notificar a todo el Jurado, incluyendo al Suplente, en caso de que el proyecto no se presentará en la evaluación correspondiente, esto lo deberán realizar con al menos dos días de anticipación a la fecha programada.
- En caso de tener menos de tres asesores, deberá dejar el espacio vacío.

INSTITUTO POLITÉCNICO NACIONAL

UNIDAD PROFESIONAL INTERDISCIPLINARIA EN
INGENIERÍA Y TECNOLOGÍAS AVANZADAS

PROYECTO TERMINAL I

"Sistema para el análisis de datos meteorológicos y de calidad del aire usando microservicios y aprendizaje automático."

Que para obtener el título de

"Ingeniero en Telemática"

Presentan:

Aparicio Espinoza Octavio Joel
Hernández Castillo Francisco
López Tirado Nadia Jaqueline

Asesores:

Dr. Miguel Félix Mata Rivera

Ciudad De México a 24 de Enero de 2023

Contenido

RESUMEN 6

ABSTRACT 7

CAPÍTULO 1 INTRODUCCIÓN 9

1.1 PLANTEAMIENTO DEL PROBLEMA 10

1.2 JUSTIFICACIÓN 12

1.3 OBJETIVO GENERAL 12

1.3.1 OBJETIVOS ESPECÍFICOS 13

1.4 ALCANCES 14

1.5 SOLUCIÓN PROPUESTA 15

1.5.1 MODULO 1: MICROSERVICIOS 16

1.5.2 MODULO 2. BASE DE DATOS. 19

1.5.3 MODULO 3. MACHINE LEARNING 20

1.5.4 TECNOLOGÍAS 22

CAPÍTULO 2 ESTADO DEL ARTE 24

2.1. OBSERVATORIO ATMOSFÉRICO UNAM 24

2.2. ESTACIÓN METEOROLÓGICA CRYSTAL MOMENTS CW101 25

2.3. MONITOREO DE ESTACIÓN METEOROLÓGICA VÍA INTERNET 26

2.4. PROTOTIPO DIDÁCTICO DE UNA ESTACIÓN METEOROLÓGICA MONITOREADA A DISTANCIA 26

2.5. DISEÑO DE UNA HERRAMIENTA HIDRO-INFORMÁTICA PARA EL ANÁLISIS DE CALIDAD DE DATOS DE ESTACIONES METEOROLÓGICAS AUTOMATIZADAS 27

CAPÍTULO 3 MARCO TEÓRICO 30

3.1 ESTACIÓN METEOROLÓGICA 30

3.2 TIPOS DE ESTACIONES METEOROLÓGICAS 30

3.3 ELEMENTOS PRINCIPALES QUE CONFORMAN UNA ESTACIÓN METEOROLÓGICA. 31

3.4 ESTACIÓN METEOROLÓGICA DAVIS VANTAGE PRO2 32

3.4.1 CARACTERÍSTICAS DE LA DAVIS VANTAGE PRO2 32

3.4.2 PRECISIÓN, RANGO Y CALIDAD DE LOS DATOS DE LA DAVIS VANTAGE PRO2 33

3.4.3 CONSOLA Y PANTALLA DE DAVIS VANTAGE PRO2 35

3.5 ARQUITECTURA DE SOFTWARE 36

3.5.1 ARQUITECTURA DE MICROSERVICIOS. 36

3.6 ARQUITECTURA MONOLÍTICA EN COMPARACIÓN CON LA ARQUITECTURA DE MICROSERVICIOS 37
3.7 BENEFICIOS DE LOS MICROSERVICIOS 38
3.8 BASE DE DATOS 40
3.8.1 MYSQL 40
3.9 INTERFAZ DE USUARIO 41
3.9.1 DISEÑO UX 41
3.9.2 DISEÑO UI 41
3.10 CLOUD COMPUTING 41
3.10.1 BASE DE DATOS DE LA NUBE 42
3.10.2 CARACTERÍSTICAS CLAVE 42
3.10.3 AMAZON WEB SERVICES AWS 42
3.11 MACHINE LEARNING 43
3.11.1 ALGORITMOS DE MACHINE LEARNING 43
CAPÍTULO 4 ANÁLISIS 45
4.1.- METODOLOGIAS ÁGILES VS TRADICIONALES 45
4.2. BASE DE DATOS 47
4.2.1 MANEJADORES DE BASE DE DATOS 49
4.3 ANALISIS DE ALGORITMOS DE MACHINE LEARNING 50
4.4. ARQUITECTURAS MONOLITICAS Y DE MICROSERVICIOS 52
4.4.1 COMPARACIÓN ENTRE LOS DIFERENTES LENGUAJES DE PROGRAMACIÓN 54
4.5. MARCOS DE TRABAJO 55
4.5.1 MARCOS DE TRABAJO PARA FRONT. 57
4.6 ANALISIS DE ALGORITMOS DE MACHINE LEARNING 58
4.7. TABLA COMPARATIVA DE ALGORIMOS DE MACHINE LEARNING 61
4.8 DATASETS 66
4.9. PRUEBAS DE CONCEPTO ESTACIÓN METEOROLÓGICA 69
4.10. PRUEBAS DE CONCEPTO AWS 73
CAPÍTULO 5 DISEÑO 78
5.2. DISEÑO DE LA BASE DE DATOS 80
5.3. DIAGRAMA DE CASOS DE USO DEL SISTEMA 81
5.4. DIAGRAMAS DE SECUENCIA 83
5.4.1 CONSULTA DASHBOARD DE DATOS 83
5.4.2 CONSULTA POR RANGOS DE TIEMPO 84

5.4.3 CONSULTA PRONÓSTICO DE TIEMPO 84
5.5 ESCENARIO DE PRUEBAS 85
5.6 MUCKUPS 86
5.7 CRONOGRAMA DE ACTIVIDADES 87
5.8 CRONOGRAMA DE ACTIVIDADES DIAGRAMA DE GANTT 91
REFERENCIAS 94

Tabla de ilustraciones

Ilustración 1. Estación Meteorológica modelo DAVIS VANTAGE PRO2 ubicada en UPIITA-IPN. 10
Ilustración 2. Mapa geográfico de la ubicación de las estaciones meteorológicas. 11
Ilustración 3. Arquitectura del sistema. 15
Ilustración 4. Arquitectura en alto nivel de microservicios del sistema. 16
Ilustración 5. Diagrama a bloques del sistema planteado vista futura UPIITA-ESCOM 20
Ilustración 6. Diagrama a bloques del sistema planteado para el análisis de la dataset de Monitoreo Atmosférico CDMX Y Observatorio Hidrológico UNAM. 21
Ilustración 7. Grafica de datos de perfiles de viento. 24
Ilustración 8. Observatorio Atmosférico UNAM. 25
Ilustración 9. Estación meteorológica Crystal Moments CW101. 25
Ilustración 10. Diagrama estación meteorológica. 26
Ilustración 11. Comparativa estación meteorológica VS OpenWeatherMap. 27
Ilustración 12. Estaciones meteorológicas de Estado de México. 28
Ilustración 13. Estación meteorológica DAVIS VANTAGE PRO2. 33
Ilustración 14. Interfaz de la DAVIS VANTAGE PRO2. 35
Ilustración 15, Consola DAVIS VANTAGE PRO2. 35
Ilustración 16. Arquitectura de microservicios. 37
Ilustración 17. Comparativa arquitectura microservicios VS arquitectura monolítica. 38

ÍNDICE DE TABLAS

Tabla 1. TABLA COMPARATIVA SQL VS NOSQL..... ¡Error! Marcador no definido.
Tabla 2. TABLA COMPARATIVA VENTAJAS Y DESVENTAJAS DE SQL Y NOSQL..... ¡Error! Marcador no definido.
Tabla 3. TABLA COMPARATIVA ENTRE MANEJADORES DE BASES DE DATOS ..... ¡Error! Marcador no definido.**49**
Tabla 4. TABLA COMPARATIVA ALGORITMOS DE MACHINE LEARNING.....53
Tabla 5. TABLA COMPARATIVA ENTRE PARÁMETROS METEOROLÓGICOS.....55
Tabla 6. TABLA COMPARATIVA DE METODOLOGÍAS DE DESARROLLO.....56
Tabla 7. TABLA COMPARATIVA ENTRE CARACTERÍSTICAS DE METODOLOGÍAS DE DESARROLLO 57
Tabla 8. TABLA COMPARATIVA ENTRE LENGUAJES DE PROGRAMACIÓN.....57
Tabla 9. TABLA COMPARATRIVA DE ARQUITECTURA MONOLITICA VS MICROSERVICIOS 538
Tabla 10. TABLA COMPARATIVA DE MARCOS DE TRABAJO.....59
Tabla 11. TABLA COMPARATIVA DE MARCOS DE TRABAJO PARA FRONT 61

RESUMEN

Se propone un sistema de análisis de datos con aprendizaje automático implementando una arquitectura de microservicios para la extracción, visualización y análisis de datos meteorológicos.

Se construirá una red de datos entre dos estaciones meteorológicas; la primera ubicada en la UPIITA-IPN y la segunda que se instalará, calibrará y configurará en ESCOM-IPN como parte de este proyecto terminal.

Adicionalmente, se extraerán y analizarán datos abiertos de calidad del aire de la CDMX con el propósito de complementar e identificar patrones o tendencia en los parámetros lluvia, presión atmosférica, temperatura y humedad de los datos obtenidos de la red de estaciones, cuando comience a medir las variables meteorológicas.

Se emplearán tecnologías web, técnicas de minería de datos y de machine learning e interfaces de comunicación; con el objetivo de permitir la consulta de datos, recepción de alertas de monitoreo y análisis exploratorio del histórico de los datos recopilados desde las estaciones meteorológicas.

También, como parte del análisis de datos, y para complementarlos con datos históricos de años anteriores, se extraerán, procesarán y almacenarán datos abiertos de calidad del aire, y de bases de datos de la Dirección de Monitoreo Atmosférico CDMX y del Observatorio Hidrológico II UNAM, estas tareas también como parte de esta propuesta, para finalmente desplegar en un dashboard web los resultados y el monitoreo de estas variables de forma automática.

ABSTRACT

A data analysis system with machine learning is proposed by implementing a microservices architecture for the extraction, visualization and analysis of meteorological data.

A data network will be built between two weather stations; the first located at UPIITA-IPN and the second to be installed, calibrated and configured at ESCOM-IPN as part of this terminal project.

Additionally, open air quality data from CDMX will be extracted and analyzed in order to complement and identify patterns or trends in the data obtained from the network of meteorological stations, when it begins to measure meteorological variables.

Web technologies, data mining and machine learning techniques and communication interfaces will be used; with the objective of allowing the consultation of data in real time, reception of monitoring alerts and exploratory analysis of the history of the data collected from the meteorological stations.

Also, as part of the data analysis, and to complement it with historical data from previous years, open air quality data will be extracted, processed and stored, and from databases of the Atmospheric Monitoring Directorate and the Hydrological Observatory II UNAM, these tasks also as part of this proposal, to finally display the results and monitoring of these variables automatically in a web dashboard.

Capítulo 1. Introducción.

CAPÍTULO 1 INTRODUCCIÓN

La meteorología es la ciencia que estudia el estado del tiempo, el medio atmosférico, los fenómenos allí producidos y las leyes que lo rigen. La necesidad de información sobre los sistemas de tiempo y cómo se desarrollan es cada vez más importante para comprender el ambiente de un lugar y su variabilidad, pues el clima forma parte importante en la cotidianidad permitiendo realizar ciertas actividades y otorgar la seguridad de que no serán interrumpidas ni se complicarán por el estado del tiempo y así tomar precauciones para realizar con éxito las actividades programadas en el día a día.

En México la meteorología ha tenido un buen desarrollo durante los últimos 20 años, con la aparición de las diferentes tecnologías y dependencias que transmiten datos meteorológicos.

Tomando en cuenta lo dicho anteriormente, la meteorología es una ciencia compleja, que requiere una observación frecuente del estado del tiempo, la investigación meteorológica se ha basado en el registro y observación del cambio que sufre la atmosfera en diferentes periodos de tiempo.

Por otra parte, la calidad del aire se refiere a la presencia en mayor o menor medida de contaminantes en la atmósfera que puedan ser nocivos para la salud humana, para el medio ambiente en su conjunto y para otros bienes de cualquier naturaleza. Se establecen índices de calidad del aire que proporcionan valores indicativos del estado de la atmósfera respecto a un contaminante en particular o a un conjunto de ellos. Estos valores se refieren a una escala definida a partir de conocimientos científicos sobre los niveles de los distintos contaminantes que pueden resultar nocivos para la salud humana, el medio ambiente, etc.

Los índices de calidad del aire (ICA) se suelen expresar en categorías a partir de los posibles valores de concentración medidos, o pronosticados por un modelo, de un contaminante en concreto, o de un conjunto de ellos.

El desarrollo científico y tecnológico permite disponer de herramientas como las estaciones meteorológicas que conforman un sistema de monitoreo climático en tiempo real y que tiene por objetivo registrar y observar un máximo de datos sobre el estado de la atmosfera, analizarlos, interpretarlos y obtener deducciones prácticas, para prever el tiempo con la máxima antelación posible, esto representa un avance trascendente porque permite generar información que apoya a la toma de decisiones en diferentes ámbitos.

Obtener datos de estaciones meteorológicas permite no solo determinar las condiciones del tiempo y tomar decisiones, sino también conocer su relación con la calidad del aire al mantener un registro histórico, con esto es posible analizar el

microclima de un área específica como en la UPIITA, ESCOM o la zona de Ticomán-Zacatenco Lindavista con mayor precisión.

Debido a esto, es importante partir de la extracción de datos de las estaciones meteorológicas antes mencionadas, procesarlos y mostrarlos a partir de una arquitectura de microservicios web con la red de estaciones de meteorología que se ubicarán en UPIITA y ESCOM podrá dar a conocer sus propios registros, esto a través de la colaboración con el Laboratorio de Inteligencia GeoEspacial y Cómputo Móvil de la UPIITA, quien proveerá el acceso a dichas estaciones. Y finalmente, en consecuencia, poder tomar mejores decisiones con respecto a actividades al aire libre, deportivas, flujos de personas y estancias, todo basándose en el análisis de datos dentro de las unidades Académicas.

1.1 PLANTEAMIENTO DEL PROBLEMA

Actualmente la UPIITA cuenta con una estación meteorológica modelo DAVIS VANTAGE PRO2 que fue instalada desde el año 2019, sin embargo, no cuenta con un módulo de software de visualización de datos adecuado a las necesidades que se requiere para análisis e investigación, ya que los datos se requieren en formatos interoperables y almacenados en una base de datos de lo cual carece la actual consola propia de la estación utilizada dentro de la unidad Académica.

Ilustración 1. Estación Meteorológica modelo DAVIS VANTAGE PRO2 ubicada en UPIITA-IPN.

Lo que puede significar un problema a la hora de realizar alguna investigación donde se necesiten estos datos, ya sea desde el estado de clima hasta parámetros más específicos como lo son: Velocidad y dirección del viento, Temperatura ambiente y humedad relativa, Lluvia, Radiación solar y Radiación UV, Presión atmosférica, Temperatura y humedad interior.

Otra problemática, es que la estación ubicada en la UPIITA, al no contar con una base de datos, solo tiene registros parciales del año 2019, lo cual no permite analizarlos históricamente o postularlos para aplicar algún algoritmo de aprendizaje automático, uno de los propósitos de este PT, es precisamente dotarlo de la capacidad de registrar los datos capturados en una base de datos de manera ininterrumpida para conformar una base de datos histórica.

La segunda estación meteorológica modelo DAVIS VANTAGE PRO2 se plantea sea instalada en agosto del 2022 en ESCOM-IPN para conformar la red de estaciones meteorológicas, la red de datos se conformará usando un equipo de cómputo que se conectara a las consolas de la estaciones y a su vez estas computadoras se conectaran vía webservices y enlazando las bases de datos que también se construirán , todas estas tareas forman parte de esta propuesta de proyecto terminal y que serán realizadas por los 3 integrantes.

Ilustración 2. Mapa geográfico de la ubicación de las estaciones meteorológicas.

Considerando que para visualizar las variables de manera precisa se requiere de un dataset enriquecido con los suficientes datos, para poder analizarlos y con ello poder aplicar algún algoritmo de aprendizaje automático para poder obtener un pronóstico o identificar una tendencia.

Así mismo no hay ningún tipo de sistema funcional implementado, ya sea con arquitectura monolítica o de microservicios para extraer, procesar y analizar la información que se obtendrá de las estaciones y es parte de los objetivos de este proyecto terminal poder conformarlo.

De igual manera todo el personal, estudiantes de UPIITA y ESCOM necesitan conocer el estado del tiempo, Velocidad y dirección del viento, Temperatura

ambiente y humedad relativa, Lluvia, Radiación solar y Radiación UV, Presión atmosférica, Temperatura y humedad interior, ya que no se ha hecho un análisis exploratorio de datos de fenómenos antropogénicos a la información recabada que permita descubrir el comportamiento del medio ambiente a nivel de superficie en la UPIITA-IPN, ESCOM ni en la zona de Ticomán-Zacatenco Lindavista. En este caso ambas unidades Académicas carecen de alguna interfaz, página web o aplicación donde se pueda visualizar la información requerida por parte de la comunidad.

Por ello, la consola actual de dicha estación no informa la obtención de datos atípicos que no corresponden al comportamiento histórico de los datos, lo cual podría ser de gran utilidad para la toma de decisiones por parte del investigador, o público en general que desee consultar esta información.

No hay implementación de Machine Learning, el cual ayuda a procesar la información obtenida de las bases de datos y reconocer patrones en ella, esto no es posible puesto que en la unidad Académica de la UPIITA-IPN ni en la unidad Académica ESCOM-IPN existe un sistema enfocado al propósito de hacer predicciones sobre variables meteorológicas y de calidad del aire.
Resumiendo, estas son las problemáticas por resolver en este proyecto terminal.

1.2 JUSTIFICACIÓN

Actualmente la UPIITA cuenta con una estación meteorológica modelo DAVIS VANTAGE PRO2 que fue instalada desde el año 2019, sin embargo, no cuenta con un módulo de software de visualización de datos adecuado a las necesidades que se requiere para análisis e investigación, ya que los datos se requieren en formatos interoperables y almacenados en una base de datos de lo cual carece la actualmente la consola propia de la estación utilizada dentro de la unidad Académica.

1.3 OBJETIVO GENERAL

Desarrollar un sistema web para la visualización y análisis de datos meteorológicos y de calidad del aire, construyendo una red entre dos estaciones meteorológicas, implementando tecnologías web, una arquitectura de microservicios, y aplicando algoritmos de aprendizaje automático para identificar tendencias o patrones de lluvia, presión atmosférica, temperatura y humedad en los datos recopilados. Desplegando datos históricos, emisión de alertas y visualización histórica de los datos.

1.3.1 OBJETIVOS ESPECÍFICOS

- Integración y diseño de las bases de datos para almacenar los datos de las estaciones meteorológicas, datos abiertos de calidad del aire y de observatorio hidrológico.

- Construcción de una API de microservicios para acceder a los datos de la red de estaciones meteorológicas.

- Desarrollar el módulo de extracción de datos desde la estación de monitoreo con una arquitectura de microservicios para poder consumirla como API.

- Desarrollar una aplicación web en la cual se observe la interfaz de visualización de datos en tiempo real, así como los datos históricos recabados de forma gráfica.

- Desarrollar el módulo para hacer la exploración de datos recabados por la estación de monitoreo.

- Desarrollar el módulo para la gestión del catálogo de fenómenos de monitoreo y alertamiento.

- Diseñar el modelo de datos que permita explorar los datos en tiempo real y recabados.

- Desarrollar el módulo de alertamiento: Permitirá gestionar un catálogo de fenómenos antropogénicos para el monitoreo en tiempo real, así como la configuración de alertas en caso de que se hallen valores atípicos en tiempo real en la estación de monitoreo (fenómenos antropogénicos).

- Desarrollar el módulo de exploración y analítica de datos recolectados: Mediante técnicas de exploración de datos para indicar valores atípicos, descubrimiento de hallazgos dentro de los datos que ya han sido recolectados, y correlaciones entre valores de meteorología y calidad del aire, así como el monitoreo de eventos antropogénicos.

Nota: en el entendido que “tiempo real” se refiere a la actualización de los datos desplegados en el dashboard que generan las estaciones meteorológicas será de 8 horas.

1.4 ALCANCES

Al finalizar el desarrollo del proyecto terminal se espera cumplir con los objetivos establecidos en este documento.

Principalmente desarrollar un sistema que permita el análisis de datos meteorológicos obtenidos desde las estaciones meteorológicas localizadas en UPIITA-IPN y ESCOM-IPN lo cual implica:

- Construir una arquitectura de microservicios de tal forma que la extracción de datos de la estación meteorológica sea un API a la que podamos consumir.

- Dentro la arquitectura de microservicios habrá modulo para el guardado de los datos de cada estación meteorológica con su respectiva base de datos montada en AWS.

- Desarrollar un dashboard para la visualización de datos, es decir crear una aplicación web para visualizar en gráficos estos datos.

- Dentro de la aplicación web habrá un apartado para consultar el comportamiento histórico de los datos recopilados el cual contendrá datos obtenidos de la Dirección de Monitoreo Atmosférico CDMX y del Observatorio Hidrológico II UNAM, así como los datos recopilados desde la misma estación meteorológica Ubicada en la UPIITA-IPN. De igual forma a modo de tabla de información, así como gráficos, para que de esta forma sea más fácil el entendimiento de los datos.

- Implementar un análisis de datos para la predicción y detección de valores meteorológicos mediante la aplicación de algoritmos de Machine Learning para encontrar patrones de lluvia, presión atmosférica, temperatura y humedad en la información recabada con el fin de hacer predicciones climatológicas como de temperatura, humedad, probabilidad de lluvia o eventos resaltables con un alto nivel de fiabilidad.

- Una vez hecho el análisis, mostrarlos de igual forma dentro de la aplicación web, para poder visualizar y poder obtener la predicción de datos esperada.

1.5 SOLUCIÓN PROPUESTA

Se propone la creación de un sistema de análisis de datos, con los siguientes módulos:

1. Modulo Microservicios: Implementación de una arquitectura de microservicios para la extracción de datos de las estaciones meteorológicas ubicadas en la UPIITA-IPN y en ESCOM-IPN. [5]

2. Modulo Base Datos MySQL en AWS: Diseño e implementación de las bases de datos para el almacenamiento histórico de datos de cada una de las estaciones, la cual se integrará con una base de datos en la nube para permitir consultas históricas desde cualquier punto.

3. Modulo Machine Learning (Aprendizaje automático): Implementación de algoritmos de machine learning para identificar patrones en las variables de lluvia, presión atmosférica, temperatura y humedad recopiladas y/o generar algún pronóstico de estas. El desarrollo de una aplicación web (dashboard) para la visualización y consulta de datos meteorológicos obtenidos desde las estaciones antes mencionadas, datos históricos y la predicción o identificación de patrones en el comportamiento de dichas variables. [6]

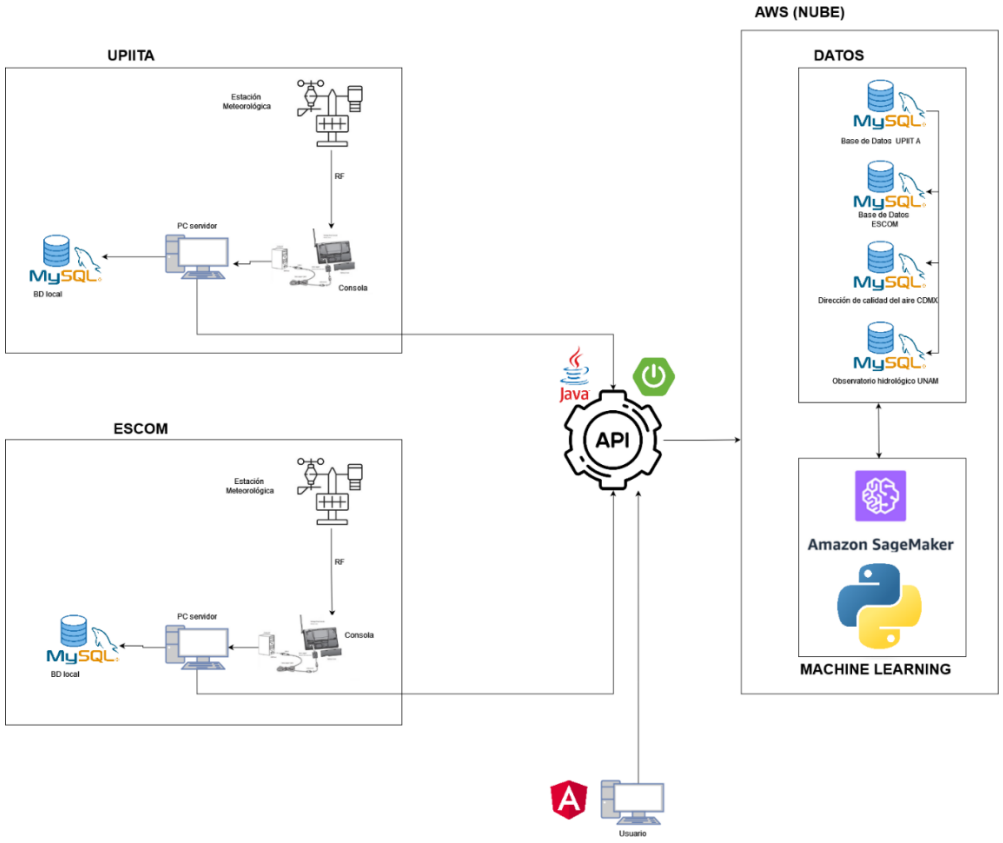


Ilustración 3. Arquitectura del sistema.

1.5.1 MODULO 1: MICROSERVICIOS

Para este módulo se debe considerar lo siguiente:

Debido a que obtendremos datos de las estaciones meteorológicas modelo DAVIS VANTAGE PRO2, se implementará una arquitectura de microservicios para la creación de dicho sistema, que contendrá módulos con las siguientes funcionalidades, en el entendido que cada funcionalidad de un microservicio es un recurso.

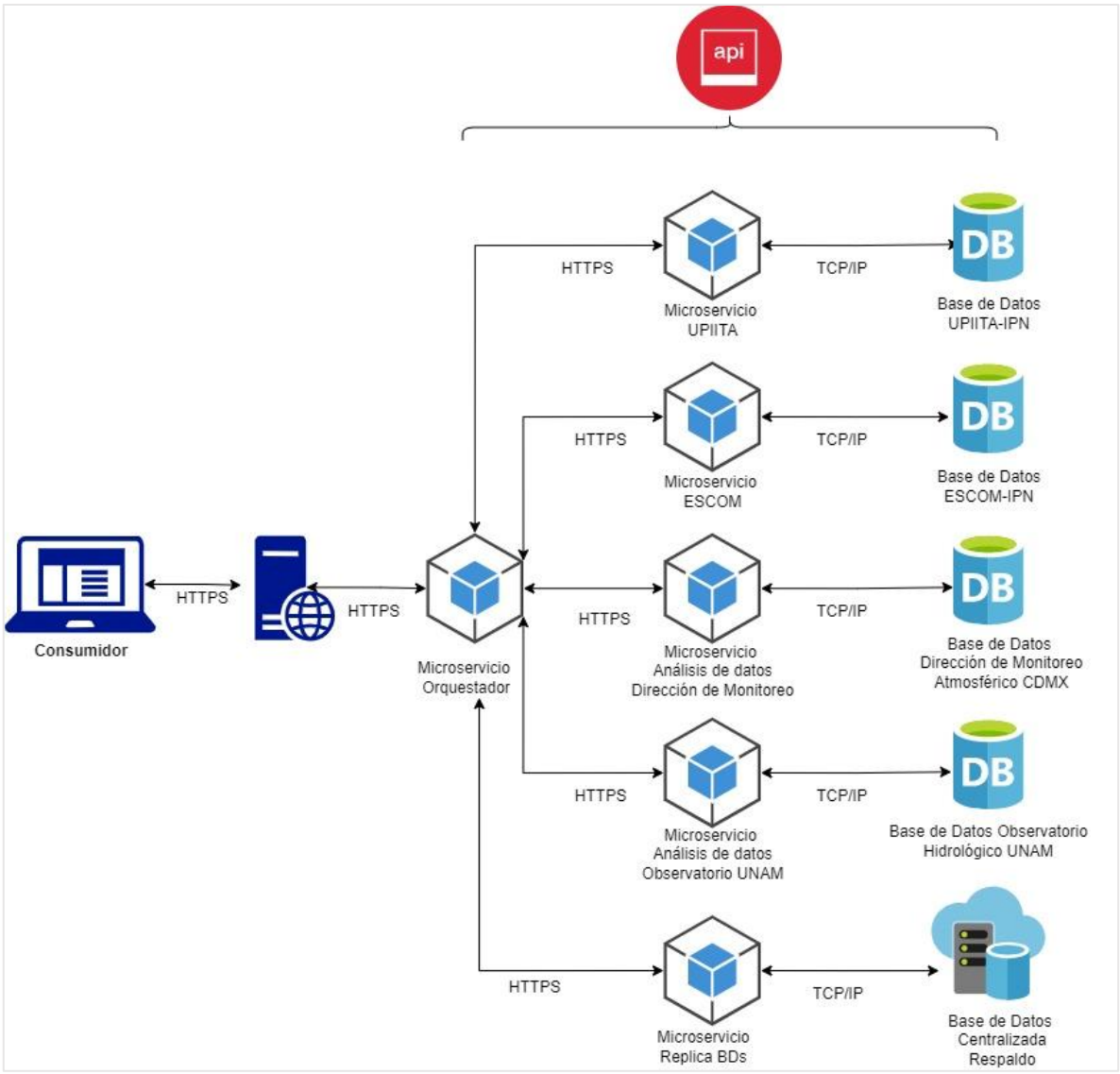


Ilustración 4. Arquitectura en alto nivel de microservicios del sistema.

El diagrama anterior de microservicios muestra a detalle el contenido dibujado dentro del componente “API” del diagrama de Arquitectura del sistema:

Microservicio Orquestador:

- Como su nombre lo indica orquestará cada uno de los llamados que se estarán realizando desde el FRONT (Aplicación web), a la funcionalidad correspondiente.

Microservicio UPIITA contará con los siguientes recursos:

- Conexión entre la estación meteorológica de UPIITA-IPN y la PC-Servidor.
- Extracción de datos de la estación meteorológica de UPIITA-IPN.
- Guardado de la recopilación de información hacia la base de datos en AWS de la estación meteorológica de UPIITA-IPN.
- Consulta y generación de dashboard o tabla de datos de la estación meteorológica de UPIITA-IPN.
- Generación de dashboard o tabla de datos de la estación meteorológica de UPIITA-IPN con la aplicación de algoritmos Machine Learning.

Microservicio ESCOM contará con los siguientes recursos:

- Conexión entre la estación meteorológica de ESCOM-IPN y la PC-Servidor.
- Extracción de datos de la estación meteorológica de ESCOM-IPN.
- Guardado de la recopilación de información hacia la base de datos en AWS de la estación meteorológica de ESCOM-IPN.
- Consulta y generación de dashboard o tabla de datos de la estación meteorológica de ESCOM-IPN.
- Generación de dashboard o tabla de datos de la estación meteorológica de ESCOM-IPN con la aplicación de algoritmos Machine Learning.

Microservicio Análisis de datos Dirección de Monitoreo.

- Conexión a la base de datos que será creada con el dataset de la Dirección de Monitoreo Atmosférico CDMX.
- Extracción de los datos de la Dirección de Monitoreo Atmosférico CDMX.
- Consulta y generación de dashboard o tabla de datos de la Dirección de Monitoreo Atmosférico CDMX.
- Generación de dashboard o tabla de datos de la Dirección de Monitoreo Atmosférico CDMX con la aplicación de algoritmos Machine Learning.

Microservicio Análisis de datos Observatorio UNAM

- Conexión a la base de datos que será creada con el dataset del Observatorio Hidrológico II UNAM.
- Extracción de los datos del Observatorio Hidrológico II UNAM.
- Consulta y generación de dashboard o tabla de del Observatorio Hidrológico II UNAM.
- Generación de dashboard o tabla de datos del Observatorio Hidrológico II UNAM con la aplicación de algoritmos Machine Learning.

Microservicio Réplica Base de Datos

- Guardado centralizado de ambas bases de datos hacia AWS.
- Aplicación de algoritmos machine learning para el análisis de los datos y predicción de estos, vista futuro.

Como observamos en la Ilustración 3. Arquitectura del Sistema, las estaciones meteorológicas se conectarán vía RF a consolas conectadas hacia computadoras, y dichas computadoras se enlazarán por medio de WIFI a la API del sistema, una vez que se logre esta conexión, cada 8 horas se realizará un guardado a sus respectivas bases de datos de cada estación y con ello un respaldo por medio de una réplica de cada una de las bases de datos hacia la nube.

¿Por qué implementar una arquitectura de microservicios?

Al implementar una arquitectura de microservicios contamos con las siguientes ventajas:

- Se pueden desarrollar y desplegar de forma independiente.
- Se pueden utilizar funciones desarrolladas por terceros.
- Se puede escalar cada módulo según sea necesario.
- Si una aplicación basada en una arquitectura orientada a microservicios falla, solo se da de baja el módulo que está fallando sin afectar a la aplicación.
- Cada microservicio se puede diseñar, desarrollar e implementar con independencia de otros microservicios. Esto aporta agilidad, dado que es más fácil implementar nuevas versiones de los microservicios con frecuencia.

Con las ventajas mencionadas anteriormente, significa que es óptimo aplicar esta arquitectura ya que, al ser dos estaciones meteorológicas, cada una hace un guardado a su respectiva base de datos y de esta forma, si llega a fallar una, la otra seguirá funcionando mientras se le hace el cambio o mantenimiento a la funcionalidad especifica en contingencia.

1.5.2 MODULO 2. BASE DE DATOS.

Los datos que se recopilarán desde las estaciones meteorológicas serán:

- Fecha
- Hora
- Temperatura.
- Velocidad y dirección del viento.
- Lluvia actual y acumulada.
- Presión atmosférica actual.
- Humedad.

Considerando las variables anteriores, se debe diseñar una base de datos que permita almacenar dichos datos de una forma organizada para su futura consulta, realización de búsquedas, nuevos registros de datos, etc. Ya que se hará una recopilación de datos(registros) cada cierto tiempo y se enviará un respaldo a una base de datos en la nube.

Una vez que se tenga la base de datos, se integraran datos de la red de calidad del aire de la CDMX (la cual cuenta con datos desde el año 1995) así mismo se integrará otro dataset del Observatorio Hidrológico de la UNAM, el cual tiene datos de eventos pluviales.

El objetivo de la integración de estos datos es robustecer la base de datos meteorológica y observar el comportamiento y posible relación entre dichas variables para identificar cual tipo de algoritmo de Machine Learning se le puede aplicar para descubrir los patrones de lluvia, presión atmosférica, temperatura y humedad, caracterizar el fenómeno o realizar algún pronóstico (La toma de decisión de esto, se hará con base en un análisis que es parte de las actividades de esta propuesta de Proyecto Terminal).

Resolviendo el tema que se menciona en el planteamiento del problema, dado a que solo hay una estación meteorológica instalada y que se perdió gran parte de información desde el 2019, la base de datos será muy pequeña para poder aplicar este análisis, por lo que se propone realizarlo con el comportamiento histórico de los datos recopilados de la Dirección de Monitoreo Atmosférico CDMX y del Observatorio Hidrológico II UNAM los cuales contendrán al menos 10 años de datos.

Es por ello por lo que se debe realizar un análisis y diseño de la base de datos, de tal forma que se guarden las variables en donde se intersectan o convergen, tanto las de las estaciones meteorológicas ubicadas en UPIITA-IPN y ESCOM-IPN, así como las de la Dirección de Monitoreo Atmosférico CDMX y del Observatorio Hidrológico II UNAM, ya que el análisis de datos será dentro de estas, pues se espera que vista futuro únicamente se aplique a los datos recabados de las estaciones meteorológicas de UPIITA-IPN y ESCOM-IPN.

1.5.3 MODULO 3. MACHINE LEARNING

Se desarrollará e implementará un dashboard o tablero de datos (gráficos y tablas) para su visualización, el cual será manipulado para beneficio del análisis de datos y así poder implementar un componente de alertamiento para detectar valores atípicos obtenidos, esto con el fin de conocer el comportamiento histórico de las variables ambientales y de calidad de aire, para ello se propone recopilar paradigmas de inteligencia artificial y determinar el que mejor se adapte a la propuesta de solución como regresión o agrupación. [7]

Esto con el fin de conocer el comportamiento histórico de las variables ambientales y de calidad de aire, es decir, supongamos que se obtienen 6 meses de datos, se podría responder ¿Cuáles son los valores mínimos y máximos por mes de PM10? ¿Cuál es la temperatura promedio mensual, bimestral o semestral?, ¿Cuál fue la dirección del viento en promedio por quincena?, etc.

Para el desarrollo de la aplicación web donde se encontrará el dashboard o tablero de datos, se requiere mencionar algunos usos, los cuales servirán como beneficio directo cuando se intenten visualizar los datos recabados:

- Permitir a los usuarios localizar información de las variables ambientales y calidad del aire, así como del clima, entre otras, de forma rápida y sencilla.

- Permitir a los usuarios localizar información de las variables ambientales y calidad del aire, así como del clima, entre otras, en el histórico que estará disponible de forma rápida y sencilla.

- Permitir la visualización del análisis de datos con los algoritmos de Machine Learning.

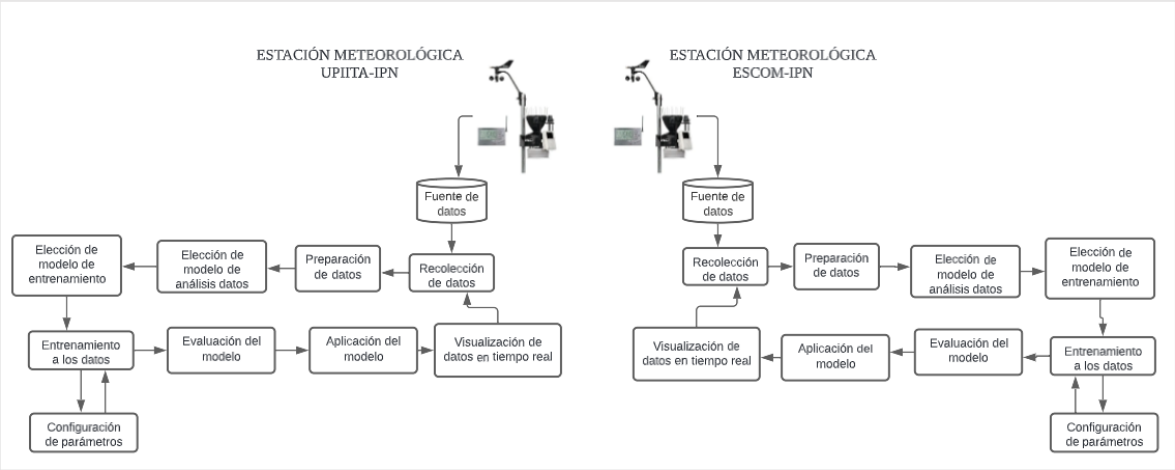


Ilustración 5. Diagrama a bloques del sistema planteado vista futura UPIITA-ESCOM

La Ilustración 5. “Diagrama a bloques del sistema planteado vista futura UPIITA-ESCOM”, es la metodología para el análisis de datos, aplica cuando existen datos y se quieren acoplar. Debido a que en la aplicación que se está planteando, al crearla, los datos serán “nuevos”, es decir, apenas se crearán los datos y se espera que, al realizar bien la aplicación, ya estén “normalizados” por lo que no se podrá usar la metodología, incluso si existen varios “actores” que la alimentan. Por lo tanto, no se podrá realizar este preprocesamiento a los datos recabados de las estaciones meteorológicas ubicadas en UPIITA y ESCOM.

No se aplicaría realmente minería, se utiliza minería cuando ya existen los datos, y como no se tienen debido a la perdida de datos de gran parte del año 2020 y 2021, como se menciona anteriormente, se realizará este análisis con la base de datos que contiene el comportamiento histórico de los datos recopilados de la Dirección de Monitoreo Atmosférico CDMX y del Observatorio Hidrológico II UNAM los cuales contendrán al menos 10 años de datos. [8]

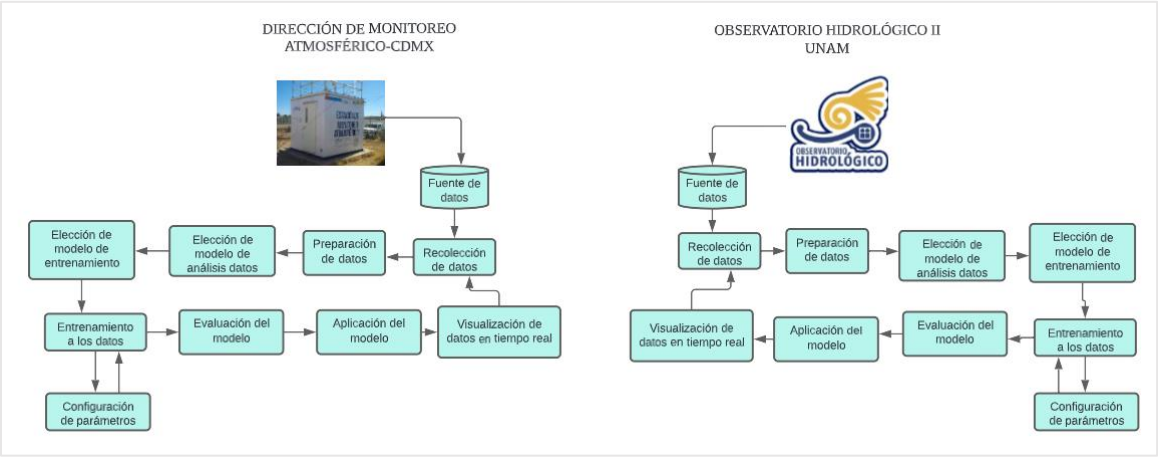


Ilustración 6. Diagrama a bloques del sistema planteado para el análisis de la dataset de Monitoreo Atmosférico CDMX Y Observatorio Hidrológico UNAM

El preprocesamiento se usa para limpiar los datos que ya existen, vengan de una o varias fuentes. De tal forma que los datos que se generarán o serán recabados de las estaciones meteorológicas ubicadas en UPIITA y ESCOM, ya estarán “limpios”.

Entonces la recolección de datos no es recolección de datos como en la Ilustración 5. “Diagrama a bloques del sistema planteado”, es “Ingreso de datos”. La integración de datos no es integración de la metodología de procesamiento de datos, es una base de datos que se diseñará. La seguridad solo usa el algoritmo y tiene un módulo de validación, porque los datos que se ingresarán y como se comenta, ya estarán limpios por lo que no hace limpieza, menos recolección e integración.

1.5.4 TECNOLOGÍAS

El uso de tecnologías o herramientas tecnológicas serán definidas en el capítulo 4 “Análisis” en el que:

- Se comparará y se seleccionará la tecnología a partir de las ventajas y desventajas entre Spring Boot JAVA, Python o lenguajes de programación para el desarrollo del BackEnd, que soporte arquitectura de microservicios.

- Se realizó una comparación entre manejadores de bases de datos (SGBD) como MySQL, SQL Server, MongoDB, MariaDB, PostgreSQL, etc., la cual consistía en quien soporte la gran cantidad de datos tanto para el registro histórico de las estaciones meteorológicas ubicadas en UPIITA y ESCOM, así como para el análisis de datos de la Dirección de Monitoreo Atmosférico CDMX y del Observatorio Hidrológico II UNAM, facilidad de uso al igual que el conocimiento previo por cada uno de los integrantes del equipo y pudimos concluir que con la investigación realizada que la tecnología y el manejador que mejor se adapta a nuestro proyecto y habilidades seria MySql.

- De igual forma se realizó un análisis de la nube como IBM Cloud, AWS, Azure, entre otras para tener ambas bases de datos de su respectiva estación meteorológica y concluimos que la tecnología que vamos a utilizar será Amazon Web Services (AWS).

- Para el desarrollo del dashboard y tabla de datos o gráficos se utilizará HTML, CSS y java scrpit con ayuda de angular.

Y finalmente se hará una comparación para definir qué algoritmo de Machine Learning es necesario como agrupación, regresión, etc., y que aplica mejor a la solución aquí planteada para poder generar la predicción de los históricos de las variables recabadas.

Capítulo 2.
Estado del Arte.

En breve presentaremos los proyectos y algunas estaciones meteorológicas que se han desarrollado hasta ahora. [12]

2.1. OBSERVATORIO ATMOSFÉRICO UNAM

Ubicada de Ciudad Universitaria en la ciudad de México se encuentra el observatorio atmosférico de la UNAM.

Esta estación nos brinda datos en tiempo real los cuales podemos descargar, así como también visualizar las gráficas de cada factor que se mide.[1]

Un ejemplo de las gráficas que nos muestran es la siguiente la cual mide los perfiles de viento: [17]

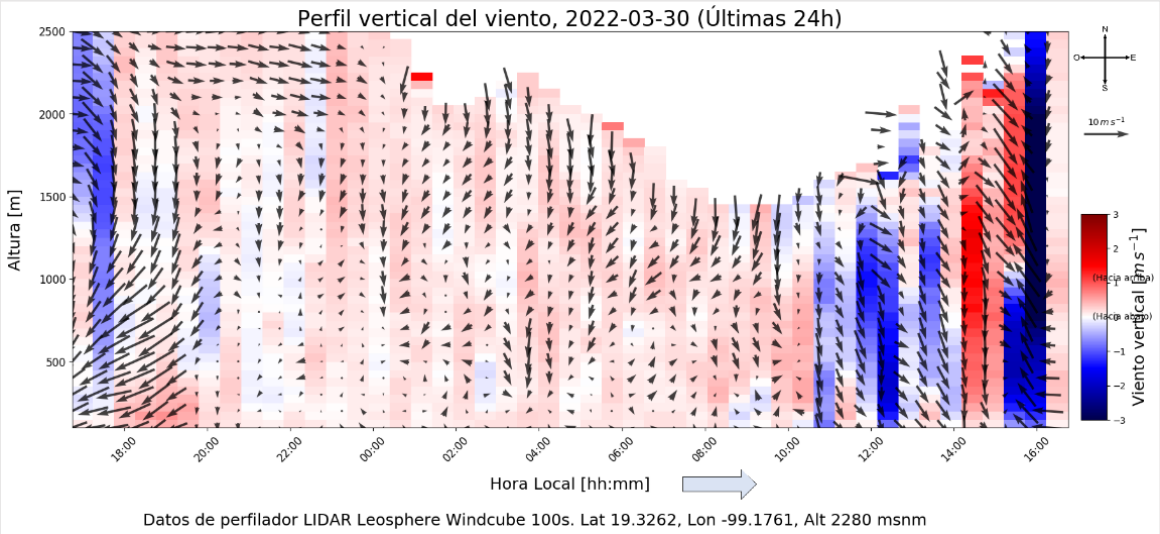


Ilustración 7. Grafica de datos de perfiles de viento.

Así como nos dan datos sobre perfiles de viento también podemos ver gases de efecto invernadero, calidad de aire, carbono negro, campo eléctrico, agua total precipitable entre algunos más. A continuación, la estación meteorológica:

Ilustración 8. Observatorio Atmosférico UNAM.

2.2. ESTACIÓN METEOROLÓGICA CRYSTAL MOMENTS CW101

La estación consta de un reloj radio controlado, panel retro iluminado y un icono del pronóstico iluminado con luz LED los cuales en conjunto y con la estación en funcionamiento nos brindan datos como lo puede ser:

- Temperatura interior 0 a +50 ° C.
- Rango de temperatura exterior de -20 a + 60 ° C.

La estación tiene una base acabada en negro y que muestra la información de tiempo y temperatura.

Ilustración 9. Estación meteorológica Crystal Moments CW101.

2.3. MONITOREO DE ESTACIÓN METEOROLÓGICA VÍA INTERNET

Es un proyecto realizado en la Universidad Autónoma de Zacatecas el cual fue hecho para que un sistema electrónico pudiera capturar datos así mismo como el poder transmitirlos y desplegarlos de una estación meteorológica.

Consistía en una estación donde realizaban las mediciones por medio de sensores conectados a un microcontrolador PIC16F877A El cual realizaba la digitalización y las capturas de las señales analógicas de los sensores. Para poder hacer la interfaz se utilizó la tarjeta NI-ELVIS desarrollando un programa en LabVIEW la cual ayudaba a interpretar la información que le llegaba a la tarjeta, así como también de crear un base de datos.

2.4. PROTOTIPO DIDÁCTICO DE UNA ESTACIÓN METEOROLÓGICA MONITOREADA A DISTANCIA

Este proyecto consistió en el diseño y construcción de una estación meteorológica la cual es monitoreada de forma remota el software para el monitoreo permite dar a conocer las variables meteorológicas medidas por la estación las cuales son temperatura humedad velocidad y dirección de viento.

A continuación, presentamos el diagrama del prototipo usado para el trabajo de tesis: [16]

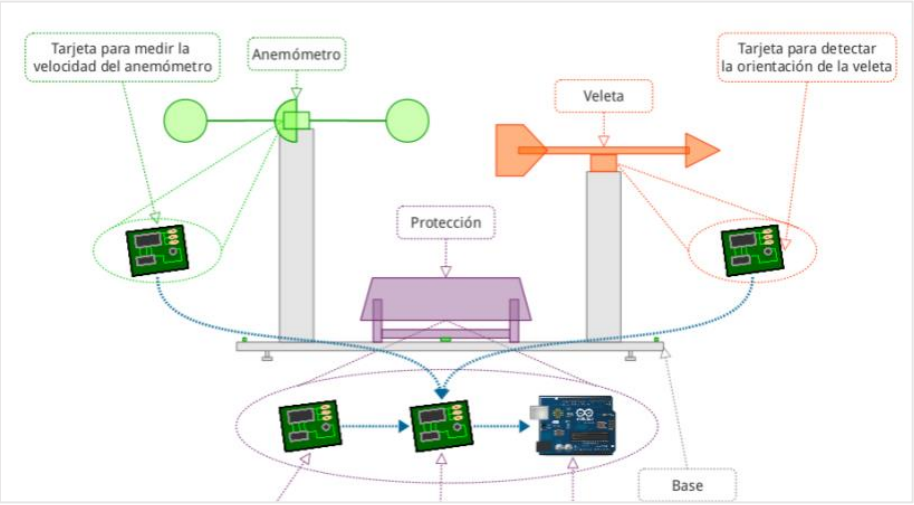


Ilustración 10. Diagrama estación meteorológica.

En seguida mostramos una gráfica comparativa de los datos obtenidos, en este caso la humedad:

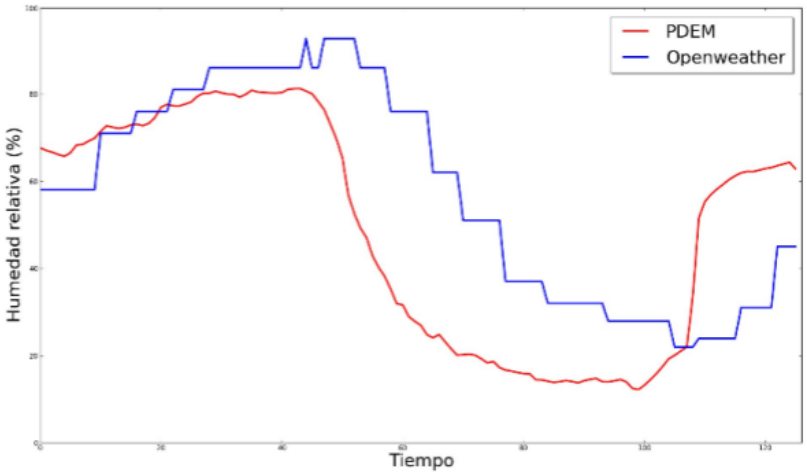


Ilustración 11. Comparativa estación meteorológica VS OpenWeatherMap.

La gráfica en azul nos muestra las mediciones obtenidas por OpenWeatherMap mientras que la gráfica en rojo nos muestra las mediciones obtenidas de la estación meteorológica.

2.5. DISEÑO DE UNA HERRAMIENTA HIDRO-INFORMÁTICA PARA EL ANÁLISIS DE CALIDAD DE DATOS DE ESTACIONES METEOROLÓGICAS AUTOMATIZADAS

Obtener datos meteorológicos confiables produce beneficios al usuario, en actividades como climatología, agricultura, etc. A nivel internacional la Organización Meteorológica Mundial (OMM) en conjunto con el Servicio Meteorológico Nacional (SMN), no cuenta con una normativa estandarizada para analizar la calidad de los datos meteorológicos antes de publicarlos.

La falta de esta normativa ha sido tema de investigación de diversos autores, proponiendo distintas metodologías de análisis, aunado a la falta de herramientas computacionales que permitan automatizar el proceso. La herramienta tiene una interfaz de fácil uso, cuya estructura permite ser aplicable tanto a datos en tiempo real, como a datos estadísticos.

Con su empleo se genera información útil para la detección de valores atípicos de los datos, alertando al usuario con una marca de calidad. Con el propósito demostrar la utilidad de la herramienta para el análisis de datos, se utilizó para verificar la calidad de información que proporcionan las estaciones meteorológicas

automáticas (EMA's) y las estaciones sinópticas meteorológicas (ESIME's) del Estado de México.

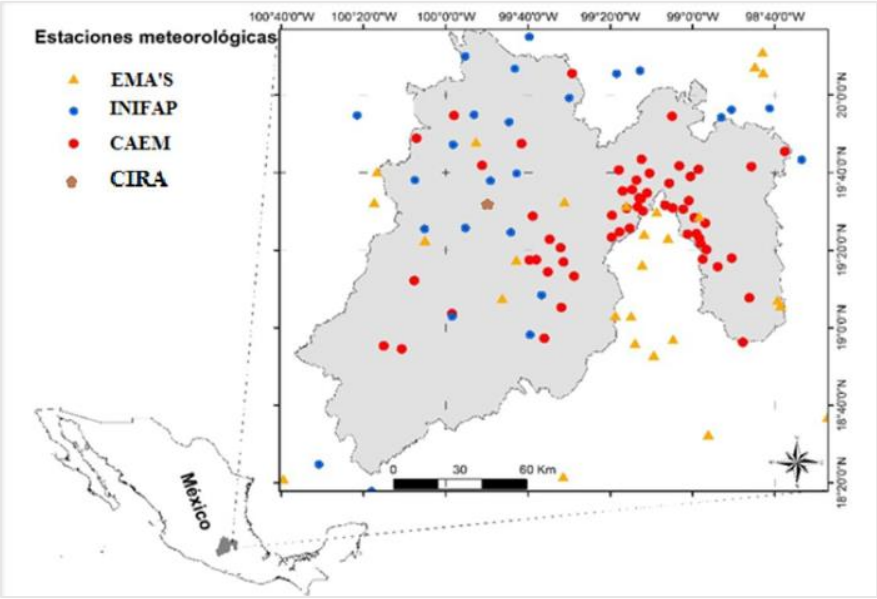


Ilustración 12. Estaciones meteorológicas de Estado de México.

Capítulo 3.
Marco Teórico.

3.1 ESTACIÓN METEOROLÓGICA

Una estación meteorológica es un conjunto de herramientas de medición que nos permite predecir el clima tanto en tiempo real como a largo-mediano plazo. El principal objetivo es recabar la información suficiente para poder predecir el clima en un tiempo próximo. La utilidad principal de una estación meteorológica es recoger y registrar datos meteorológicos, con esos datos se crea información con muchas funcionalidades como lo son: Saber exactamente las condiciones meteorológicas de ese ese lugar, comparar esa información con otras estaciones meteorológicas de lugares cercanos, aportar información para realizar los pronósticos meteorológicos de los modelos numéricos, crear información climática representativa del lugar en donde se toman los datos entre muchas más funciones.

Dependiendo la estación meteorológica que utilicemos y la cantidad de herramientas que dispongamos seremos capaces de medir varios parámetros los cuales nos sirven para pronosticar el clima, entre los más básicos son la humedad y temperatura, pero también podemos medir presión barométrica, velocidad del viento, dirección del viento, precipitaciones, temperatura en suelo, humedad del suelo, radiación solar por decir algunas.

3.2 TIPOS DE ESTACIONES METEOROLÓGICAS

Estaciones meteorológicas domésticas digitales: y son normalmente aquellas destinadas para el público en general y tienen un costo relativamente, un modelo de este tipo de estaciones seria Estación meteorológica Bresser 7007320 el cual es capaz de recabar datos como lo son temperatura, humedad y presión atmosférica. Al ser modelos bastantes básicos no disponen de una interfaz sofisticada y tampoco de puertos de entradas y salidas.

Estaciones meteorológicas con conexión a PC: estas tienen la capacidad de concertarse a una computadora y también almacenar datos y poderlos exportar en herramientas como lo son Excel y permiten visualizar los datos meteorológicos en un software. Algunas características de esta son: información meteorológica común como temperatura, humedad, presión atmosférica, velocidad y dirección del viento, precipitaciones. También cuenta con una consola central con conector USB capaz de transmitir datos a un software de PC. Son capaces de almacenar datos puestos que disponen de una memoria interna y pantallas donde nos muestran más parámetros. Un modelo podría ser el Froggit WH1080.

Estaciones meteorológicas Wifi: Este tipo de estaciones son las que son capaces de transmitir datos a Internet y emitirlos online. Su conexión bien puede ser por wifi, por cable al modem etc. Una característica que nos ofrece es información meteorológica común: Temperatura, humedad, presión atmosférica, velocidad y dirección del viento.

Estaciones meteorológicas profesionales automáticas: Este tipo de estaciones son las que usan los servicios meteorológicos oficiales para registrar y transmitir los datos de áreas remotas o sin atención. Miden mayos precisión parámetros temperatura, humedad relativa, presión atmosférica, viento, precipitación y radiación. También disponen de su propio sistema de energía como lo pueden ser paneles solares y baterías con bastante duración.

Estaciones meteorológicas analógicas: son aquellas que no necesariamente funcionan con energía eléctrica por lo tanto todos sus instrumentos son o bien mecánicos, manuales o funcionan a través de fluidos que cambian sus propiedades por la influencia de las variables meteorológicas como la temperatura, la humedad o la presión atmosférica. [11]

3.3 ELEMENTOS PRINCIPALES QUE CONFORMAN UNA ESTACIÓN METEOROLÓGICA.

- Termómetro: El instrumento que la temperatura en el aire y algún as tipos de termómetros pueden ser el termómetro Six-Bella ni que puede ser de alcohol o mercurio, termómetros de máximas y mínimas que únicamente registran la máxima y mínima temperatura, termómetro húmedo.

- Higrómetro: Puede mide la temperatura y humedad ambiental y calcula el punto de rocío donde el punto de rocío es la temperatura a la cual se debe enfriar el aire para que el vapor de agua se condense en rocío o escarcha.

- Pluviómetro: Mide la cantidad de agua caída ya sea en forma de lluvia nieve o granizo.

- Anemómetro: Este es el encargado de medir la velocidad de aire.

- Veleta: La veleta es la encargada de indicarnos en qué dirección va el viento y trabaja en conjunto con el anemómetro.

- Barómetro: Mide la presión atmosférica y nos da información sobre si nos encontramos en altas o bajas presiones. Es un gran indicador de la evolución del tiempo que nos puede dar información sobre cómo está evolucionando y a que nos podemos ver expuestos.

- Termógrafo: Sus funciones es similar al de un sismógrafo, hay un bolígrafo que está en permanente contacto con una hoja de papel calibrada con las temperaturas y que va moviéndose en función del tiempo.

- Piranómetro: Mide la radiación solar global.

- Heliógrafo: Este dispositivo es el que mide cuantas horas de luz hay en el día. [9]

3.4 ESTACIÓN METEOROLÓGICA DAVIS VANTAGE PRO2

Algunas de las variables que proporciona son:

- Temperatura y humedad interior y exterior.
- Temperatura de sensación y punto de rocío.
- Velocidad y dirección del viento.
- Lluvia actual y acumulada diaria, mensual y anual.
- Intensidad de lluvia.
- Presión atmosférica actual y tendencia.
- Radiación solar.
- Evapotranspiración.
- Radiación e índice UV.
- Pronóstico meteorológico.
- Fase lunar y hora de puesta y salida del sol.

Donde también incluye sensores extra como lo son:

- Sensor de humectación de hoja.
- Sonda de humedad del suelo.
- Sonda de temperatura multipropósito.

3.4.1 CARACTERÍSTICAS DE LA DAVIS VANTAGE PRO2

- Sensores separados: posibilidad de colocarlos en diferentes. Ofrece la mejor calidad de datos ya que no se interfieren entre ellos. Esto permite también en caso de ruptura de alguno de los sensores sustituirlos o repararlo por separado.
- Si bien ya cuenta con sensores tenemos la posibilidad de Posibilidad de añadir más los cuales pueden ser sensores de temperatura, temperatura – humedad y sensores de humedad del suelo y de humedad en las hojas.

- Se pueden añadir sensores UV Sensor y de radiación solar.
- Red de sensores. Se pueden instalar sensores separados entre sí hasta 300 metros para conectar granjas de sensores diferentes.

3.4.2 PRECISIÓN, RANGO Y CALIDAD DE LOS DATOS DE LA DAVIS VANTAGE PRO2

Ilustración 13. Estación meteorológica DAVIS VANTAGE PRO2.

Presión barométrica: [4]

- Resolución: 0.01» Hg; 0.1 mm; 0.1 hPa; 0.1 mb.
- Rango:16» to 32.5» Hg 410 to 820 mm 540 to 1100 hPa 540 to 1100 mb.
- Precisión: 0.03» Hg, 0.8 mm Hg, 1.0 hPa, 1.0 mb.

Tendencia barométrica:

- Resolución: 0.01» Hg; 0.1 mm; 0.1 hPa; 0.1 mb.
- Rango: 5 posiciones de flecha: crece rápidamente, crece lentamente, estable, cae lentamente, cae rápidamente.

Evotranspiración:

- Resolución: 0.01»; 0.1 mm.
- Rango: Diariamente a 32.67”; 832.1 mm Mensual y anualmente a 199.99”; 1999.9 mm.
- Precisión: Mayormente de 5% o 0.01»; 0.25 mm.

Humedad Exterior:

- Resolución: 1%.
- Rango: 1 to 100%.
- Precisión: 3% RH; 4% por encima 90%.

Punto de Roció:

- Resolución: 1ºC.
- Rango: -76℃ to +54C.
- Precisión: 1.5°C.

Radiación solar:

- Resolución: 1W/m2.
- hasta 1.800 w/m2.
- Precisión: 5% en escala completa.

Temperatura exterior:

- 0,1ºC.
- De -40ºC hasta 65ºC.
- Precisión +/- 0,5ºC.

Índice de calor:

- Resolución: 1ºC.
- -40ºC hasta 74ºC.
- Precisión: +/- 1,5ºC.

Velocidad del viento:

- Rango: 0 a 322km/h.
- Precisión: 5% o 3km/h.

Longitud de transmisión, rapidez y fiabilidad de datos en DAVIS VANTAGE PRO2:

- Velocidad de viento: 2,5 a 3 segundos.
- Dirección del viento: 2,5 a 3 segundos.
- Acumulación de lluvia: de 20 a 24 segundos.
- Ratio de precipitación: de 20 a 24 segundos.
- Temperatura exterior: de 10 a 12 segundos.
- Humedad exterior: de 50 segundos a 1 minuto.
- Radiación ultravioleta de 50 segundos a 1 minuto.
- Radiación solar de 50 segundos a 1 minuto.

3.4.3 CONSOLA Y PANTALLA DE DAVIS VANTAGE PRO2

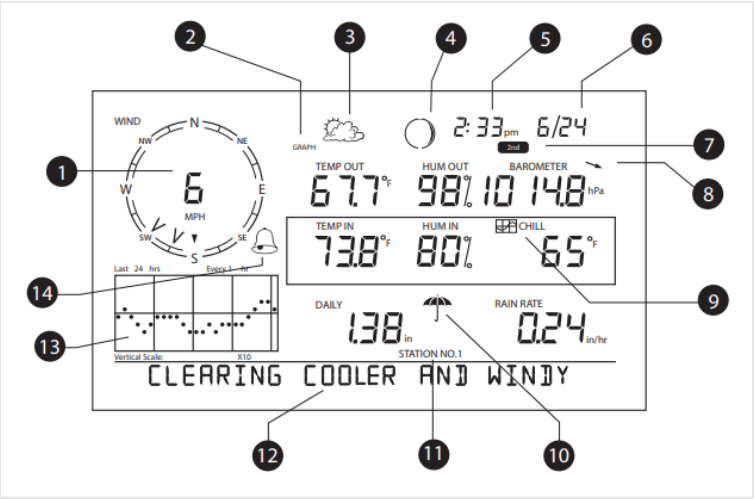


Ilustración 14. Interfaz de la DAVIS VANTAGE PRO2.

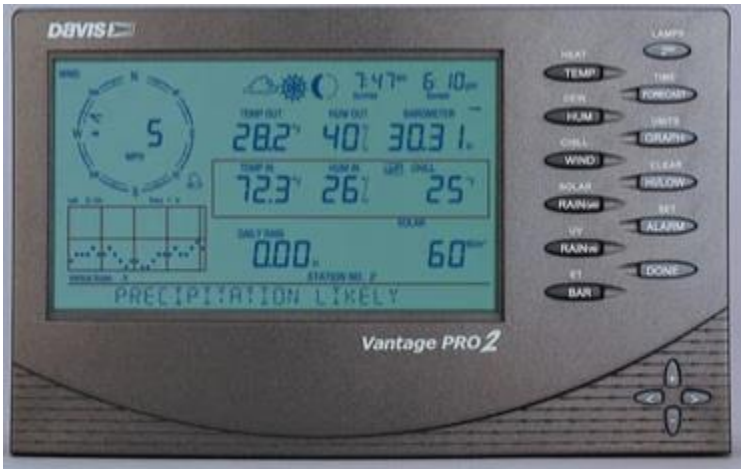


Ilustración 15, Consola DAVIS VANTAGE PRO2.

1. Rosa de los Vientos con información de la velocidad del viento y la dirección.
2. Indicador de visualización de gráfico, máximas-mínimas y modo configuración.
3. Icono de pronóstico.
4. Indicador de fase lunar.
5. Hora.
6. Fecha.
7. Indicador de botón 2ND en uso.

8. Flecha con tendencia barométrica.

9. Icono de gráfico.

10. Icono de indicador de lluvia.

11. Indicador de numero de estación.

12. Panel con información meteorológica.

13. Campo con gráficos.

14. Icono de alarma. [3]

3.5 ARQUITECTURA DE SOFTWARE

Es la manera en la que se estructura un sistema y se defina la relación entre las diferentes entidades, esta arquitectura es determinada básicamente por criterios de calidad como la modificabilidad, mantenibilidad, seguridad y el rendimiento. El mantenimiento es esencial porque, aunque un software se pueda desarrollar en unas pocas semanas o meses, lo más probable es que se mantenga durante años, añadiendo nuevas funcionalidades requeridas, o corrigiendo problemas existentes. El crecimiento también es fundamental, porque todo software cuya funcionalidad no se amplíe o se modifique, tiende a ser inservible en un relativamente corto espacio de tiempo.

3.5.1 ARQUITECTURA DE MICROSERVICIOS.

Los microservicios (o la arquitectura de microservicios) son un enfoque arquitectónico nativo en la nube en el que una única aplicación se compone de muchos componentes o servicios más pequeños, acoplables e implementables de forma independiente. Estos servicios normalmente poseen su propio recurso de tecnología, incluyendo base de datos y modelo de gestión de datos; se comunican entre sí a través de una combinación de API REST, streaming de eventos y message brokers; y están organizados por capacidad de negocio, con la línea que separa los servicios a menudo denominados como un contexto limitado.

Los microservicios son tanto un estilo de arquitectura como un modo de programar software. Con los microservicios, las aplicaciones se dividen en sus elementos más pequeños e independientes entre sí. A diferencia del enfoque tradicional y monolítico de las aplicaciones, en el que todo se compila en una sola pieza, los microservicios son elementos independientes que funcionan en conjunto para llevar a cabo las mismas tareas. Cada uno de esos elementos o procesos es un microservicio.

Este enfoque de desarrollo de software valora el nivel de detalle, la sencillez y la capacidad para compartir un proceso similar en varias aplicaciones. Es un elemento fundamental de la optimización del desarrollo de aplicaciones hacia un modelo nativo de la nube.

Sin embargo, el mayor interrogante es cuáles son las ventajas de utilizar una infraestructura de microservicios, el objetivo es distribuir software con mayor rapidez. Si bien esto se puede lograr con los microservicios, se deben considerar otras cuestiones. Dividir las aplicaciones en microservicios no es suficiente; es necesario administrarlos, coordinarlos y gestionar los datos que crean y modifican.

La diferencia entre los microservicios y la arquitectura monolítica es que los microservicios componen una única aplicación de muchos servicios más pequeños, poco acoplados en contraposición al enfoque monolítico de una aplicación grande y bien acoplada. [10]

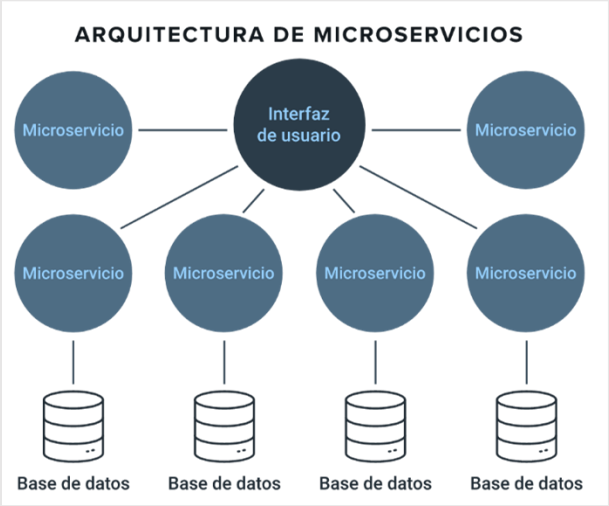


Ilustración 16. Arquitectura de microservicios.

3.6 ARQUITECTURA MONOLÍTICA EN COMPARACIÓN CON LA ARQUITECTURA DE MICROSERVICIOS

Con las arquitecturas monolíticas, todos los procesos están estrechamente asociados y se ejecutan como un solo servicio. Esto significa que, si un proceso de una aplicación experimenta un pico de demanda, se debe escalar toda la arquitectura. Agregar o mejorar las características de una aplicación monolítica se vuelve más complejo a medida que crece la base de código. Esta complejidad limita la experimentación y dificulta la implementación de nuevas ideas. Las arquitecturas monolíticas aumentan el riesgo de la disponibilidad de la aplicación porque muchos procesos dependientes y estrechamente vinculados aumentan el impacto del error de un proceso.

Con una arquitectura de microservicios, una aplicación se crea con componentes independientes que ejecutan cada proceso de la aplicación como un servicio. Estos servicios se comunican a través de una interfaz bien definida mediante API ligeras. Los servicios se crean para las capacidades empresariales y cada servicio desempeña una sola función. Debido a que se ejecutan de forma independiente, cada servicio se puede actualizar, implementar y escalar para satisfacer la demanda de funciones específicas de una aplicación.

En las aplicaciones monolíticas, con el paso del tiempo se da la tendencia de que las dependencias del código se acaben enredando. Para agregar una característica nueva, es preciso modificar el código en muchos lugares. Al no compartir el código ni los almacenes de datos, la arquitectura de microservicios minimiza las dependencias y resulta más fácil agregar nuevas características.

Al verse afectado solo un microservicio, es mucho más fácil realizar actualizaciones del esquema. En una aplicación monolítica, las actualizaciones del esquema pueden ser muy complicadas, ya que las distintas partes de la aplicación pueden tocar los mismos datos, por lo que realizar modificaciones en el esquema resulta peligroso. [18]

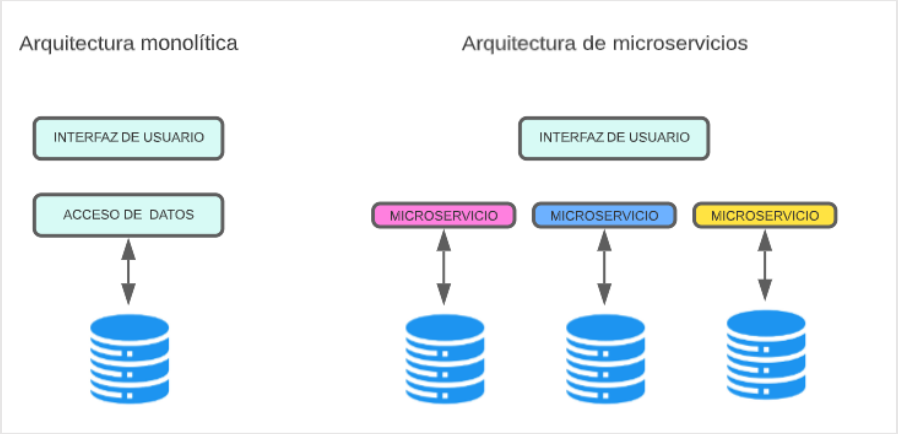


Ilustración 17. Comparativa arquitectura microservicios VS arquitectura monolítica.

3.7 BENEFICIOS DE LOS MICROSERVICIOS

- Agilidad

Los microservicios fomentan una organización de equipos pequeños e independientes que se apropian de los servicios. Los equipos actúan en un contexto pequeño y bien comprendido, y están facultados para trabajar de forma más independiente y rápida. Esto acorta los tiempos del ciclo de desarrollo. Usted se beneficia significativamente del aumento de rendimiento de la organización.

- Escalado flexible

Los microservicios permiten que cada servicio se escale de forma independiente para satisfacer la demanda de la característica de la aplicación que respalda. Esto permite a los equipos adecuarse a las necesidades de la infraestructura, medir con precisión el costo de una característica y mantener la disponibilidad si un servicio experimenta un aumento en la demanda.

- Implementación sencilla

Los microservicios permiten la integración y la entrega continua, lo que facilita probar nuevas ideas y revertirlas si algo no funciona. El bajo costo de los errores permite experimentar, facilita la actualización del código y acelera el tiempo de comercialización de las nuevas características.

- Libertad tecnológica

Las arquitecturas de microservicios no siguen un enfoque de "diseño único". Los equipos tienen la libertad de elegir la mejor herramienta para resolver sus problemas específicos. Como consecuencia, los equipos que crean microservicios pueden elegir la mejor herramienta para cada trabajo.

- Código reutilizable

La división del software en módulos pequeños y bien definidos les permite a los equipos usar funciones para diferentes propósitos. Un servicio escrito para una determinada función se puede usar como un componente básico para otra característica. Esto permite que una aplicación arranque por sí sola, ya que los desarrolladores pueden crear nuevas capacidades sin tener que escribir código desde cero.

- Resistencia

La independencia del servicio aumenta la resistencia de una aplicación a los errores. En una arquitectura monolítica, un error en un solo componente puede provocar un error en toda la aplicación. Con los microservicios, si hay un error en todo el servicio, las aplicaciones lo manejan degradando la funcionalidad sin bloquear toda la aplicación.

- Versátil

Se puede utilizar diferentes lenguajes de programación al igual que tecnologías permitiendo así adaptarse a la tecnología más adecuada y rentable que se requiera.

- Pequeño

Un microservicio debe ser lo suficientemente pequeño como que para que únicamente un único equipo lo pueda compilar e implementar porque al ser un

equipo grande tiende a ser menos productivo y aumenta la sobrecarga, la comunicación es más lenta y la agilidad disminuye.

- Apartado de errores

Si algún microservicio individual no está funcionando o no está disponible, no se interrumpe toda la aplicación siempre y cuando estén correctamente diseñados los microservicios de nivel superior para poder controlar errores. [19]

3.8 BASE DE DATOS

Debido a que en este proyecto se trabajará con una base de datos es bueno definir que una base de datos es una colección organizada de información estructurada, o datos, típicamente almacenados electrónicamente en un sistema de computadora. Una base de datos es usualmente controlada por un sistema de gestión de base de datos (DBMS).

En conjunto, los datos y el DBMS, junto con las aplicaciones que están asociados con ellos, se conocen como un sistema de base de datos, que a menudo se reducen a solo base de datos.

Existen muchos tipos de base de datos, de las más usadas son:

- Bases de datos relacionales. Los elementos de una base de datos relacional se organizan como un conjunto de tablas con columnas y filas. La tecnología de base de datos relacional proporciona la manera más eficiente y flexible de acceder a información estructurada.

- Bases de datos orientadas a objetos. La información se representa en forma de objetos, como en la programación orientada a objetos.

- Bases de datos distribuidas. Consta de dos o más archivos ubicados en diferentes sitios. La base de datos puede almacenarse en múltiples computadoras, ubicadas en la misma ubicación física o dispersas en diferentes redes.

3.8.1 MYSQL

MySql es una herramienta de gestión de base de datos siendo esta propiedad de Oracle. MySQL es de código abierto basado en el lenguaje de consulta estructurado (SQL). Funciona para sistemas operativos como Linux, UNIX y Windows. Aunque puede utilizarse en una amplia gama de aplicaciones, MySQL se asocia más a menudo con las aplicaciones web y la publicación en línea. [20]

3.9 INTERFAZ DE USUARIO

Es el medio a través del cual el administrador puede acceder a la información de la base de datos por lo cual es importante que sea amigable e intuitiva, es por ello por lo que se aplicaran los siguientes conceptos y tecnologías para la creación de esta.

3.9.1 DISEÑO UX

UX (User Experience) hace referencia a la forma en la que los usuarios interactúan con un producto o servicio. Es decir, cómo y para qué un usuario utiliza un objeto o interactúa con una web o app. Sin duda, la esencia del diseño UX está en el conocimiento de los usuarios. En otras palabras, para crear un buen diseño UX hay que comprender las necesidades de los usuarios y, por supuesto, satisfacerlas de una forma simple y clara. Por lo tanto, un buen resultado es aquel que es útil para el usuario.

3.9.2 DISEÑO UI

El diseño UI (User Interface) se centra en la parte visual. Es decir, si UX se encarga de que un producto sea útil para los usuarios, UI lo hace atractivo y visual. Se refiere a la interfaz real de un producto; el diseño visual de las pantallas por las que navega un usuario cuando usa una aplicación móvil, o los botones en los que hace clic cuando navega por un sitio web. El diseño de la interfaz de usuario se ocupa de todos los elementos visuales e interactivos de una interfaz de producto, cubriendo todo, desde la tipografía y las paletas de colores hasta las animaciones y los puntos de contacto de navegación (como botones y barras de desplazamiento).

3.10 CLOUD COMPUTING

La computación en la nube es la distribución de recursos de TI bajo demanda a través de Internet mediante un esquema de pago por uso. En lugar de comprar, poseer y mantener servidores y centros de datos físicos, puede acceder a servicios tecnológicos, como capacidad informática, almacenamiento y bases de datos, en función de sus necesidades a través de un proveedor de la nube.

3.10.1 BASE DE DATOS DE LA NUBE

Una base de datos en la nube es un servicio de base de datos creado y accedido por medio de una plataforma en la nube. Sirve a muchas de las mismas funciones que una base de datos tradicional, con la flexibilidad añadida de la computación en la nube. Para implementar la base de datos, los usuarios instalan el software en una infraestructura en la nube.

3.10.2 CARACTERÍSTICAS CLAVE

- Un servicio de base de datos creado y accedido a través de una plataforma en la nube.
- Permite a los usuarios de empresa alojar bases de datos sin comprar hardware dedicado.
- Puede ser gestionado por el usuario u ofrecido como un servicio y gestionado por un proveedor.
- Puede dar soporte a las bases de datos SQL (incluso MySQL) o NoSQL.
- Se ha accedido por medio de una interfaz web o una API proporcionada por el proveedor.

3.10.3 AMAZON WEB SERVICES AWS

Amazon Web Services, también conocida como AWS, es un conjunto de herramientas y servicios de cloud computing de Amazon. Este servicio se lanzó oficialmente en 2006 y para junio de 2007 AWS ya contaba con una base de usuarios de aproximadamente 180 mil personas. Entre las empresas que la utilizan se encuentran algunas como Reddit, Foursquare, Pinterest, Netflix, la NASA o la CIA, y algunas españolas como Mapfre, el FC Barcelona o Interflora. Esto se debe principalmente a la madurez del servicio frente a otros similares y las posibilidades que ofrece el amplio abanico de herramientas disponibles. En la Guía de Cloud Computing podrá encontrar una comparativa de todas las herramientas de Amazon Web Services con las de otras plataformas similares. [22]

3.11 MACHINE LEARNING

Es una disciplina del campo de la Inteligencia Artificial que, a través de algoritmos, dota a los ordenadores de la capacidad de identificar patrones en datos masivos y elaborar predicciones (análisis predictivo). Este aprendizaje permite a los computadores realizar tareas específicas de forma autónoma, es decir, sin necesidad de ser programados. [15]

3.11.1 ALGORITMOS DE MACHINE LEARNING

Los algoritmos de Machine Learning se dividen en tres categorías, siendo las dos primeras las más comunes: [21]

- Aprendizaje supervisado: estos algoritmos cuentan con un aprendizaje previo basado en un sistema de etiquetas asociadas a unos datos que les permiten tomar decisiones o hacer predicciones.

- Aprendizaje no supervisado: estos algoritmos no cuentan con un conocimiento previo. Se enfrentan al caos de datos con el objetivo de encontrar patrones que permitan organizarlos de alguna manera

- Aprendizaje por refuerzo: su objetivo es que un algoritmo aprenda a partir de la propia experiencia. Esto es, que sea capaz de tomar la mejor decisión ante diferentes situaciones de acuerdo con un proceso de prueba y error en el que se recompensan las decisiones correctas.

Capítulo 4.
ANÁLISIS.

Es este capítulo se realizó una amplia investigación referente a modelos de trabajo, lenguajes de programación, marcos de trabajo, bases de datos, patrones de diseño, computación en la nube, etc. para poder hacer una elección correcta y satisfactoria para la implementación y desarrollo de este proyecto, por lo que en las siguientes secciones veremos la comparación y elección de estos, así como el análisis de los requerimientos.

4.1.- METODOLOGIAS ÁGILES VS TRADICIONALES

Los marcos de trabajo (frameworks) y metodologías ágiles se caracterizan por ser adaptativas y flexibles, esto significa que no son reticentes a los cambios, al revés, los imprevistos son eventos esperados que aprenderás a acoger con gran normalidad. Hoy en día, el marco de trabajo ágil más utilizado y conocido es Scrum, pero recuerda que existen también Kanban, Lean y otros.

METODOLOGÍAS ÁGILES VS TRADICIONALES		
	Metodologías ágiles	**Metodologías tradicionales**
Definición	Sistemas especificables, se desarrollan mediante un planificación detallada y extendida	Desarrollo de código de alta calidad por pequeños equipos mediante el principio de mejora continua, basados en respuestas rápidas
Gestión	Comando y control	Liderazgo y colaboración
Modelo de desarrollo	Modelos de ciclo de vida	Modelo evolutivo de entrega
Estructura organizacional	Mecánico dirigido a grandes corporaciones	Orgánico dirigido a pequeñas y medianas corporaciones
Control de calidad	•Planificación difícil y control estricto. •Pruebas difíciles y tardías.	•Control permanente de requisitos y soluciones. •Pruebas permanentes.
Costo de reinicio	Alto	Bajo
Pruebas	Al completar la codificación	En cada iteración
Escala	Gran escala	Baja y mediana escala
Desarrolladores	•Capacidades adecuadas. •Acceso a conocimientos externos.	•Conocimientos avanzados y cooperativos.

Clientes	•Acceso a conocimiento cooperativo. •Representativo.	•Informado, cooperativo. •Capacitado.
Requerimientos	Conocidos de antemano	con cambios rápidos
Tamaño	Grandes equipos y proyectos	Pequeños equipos y proyectos.
Objetivos	Alta seguridad	Valor rápido

Tabla 1. TABLA COMPARATIVA DE METODOLOGÍAS DE DESARROLLO

Metodologías de desarrollo de software				
Características	**Kanban**	**Scrum**	**Lean**	**XP**
Enfoque	Desarrollo incremental	Iterativo, incremental	Iterativo, incremental	Iterativo, incremental
Periodo de Iteración	Empieza en el momento en que un nuevo elemento entra en la etapa "En proceso" y alguien está trabajando en él hasta que entre en la columna "Hecho".	2-4 semanas	Cuando se reciba la retroalimentación	1-6 semanas
Tamaño de proyecto	Proyectos grandes	Proyectos grandes	Proyectos grandes	Proyectos pequeños y sencillos
Complejidad de proyecto	Complejos	Complejos	Complejos	Sencillos
Prácticas	No premia la rapidez, si no la calidad del producto	Reuniones SCRUM	Aprendizaje a partir de retroalimentación	Programación en pares
Documentación	Documentación básica	Documentación básica	Documentación básica	Documentación básica
Desarrollo de características concurrentes	Posible	Posible	Posible	Posible

Tabla 2. TABLA COMPARATIVA ENTRE CARACTERÍSTICAS DE METODOLOGÍAS DE DESARROLLO

La metodología que nosotros elegimos como estrategia para la elaboración de nuestro trabajo aquí presentado es SCRUM, debido a que se debe tener una organización y control de lo que tenemos como backlog, hacer reuniones diarias para revisar el estatus de cada integrante y realizar sesiones quincenales en las que se entregue o muestre la entrega de lo que se comprometió en los inicios de sprint (planning).

4.2. BASE DE DATOS

Parte fundamental de nuestro proyecto es el almacenamiento de datos de las variables meteorológicas de estudio y que se obtienen a partir de las estaciones meteorológicas, es por ello por lo que se realizó un análisis entre bases de datos SQL y NoSQL, en las siguientes secciones encontraremos dicha información y comparación entre tablas para los diferentes manejadores de la base de datos.

SQL (Structure Query Language)

Se utiliza para definir, gestionar y manipular la información contenida en una base de datos, sql es declarativo lo que significa que tan sólo deberemos indicar el sistema de gestión de base de datos que es lo que queremos obtener que el sistema decidirá cómo obtener la información requerida. Es un lenguaje sencillo y potente que se emplea para la gestión de la base de datos a distintos niveles de utilización usuarios programadores y administradores de base de datos.

Sistemas de gestión de base de datos

Es un conjunto de programas que permiten la gestión de base de datos y realiza las funciones de extraer, almacenar y modificar información de una base de datos que donde podemos realizar acciones como eliminar modificar y analizar la información.

Características principales de un gestor de base de datos:

1. Nos permite una vista clara y centralizada para el acceso de los datos de forma segura y de la mejor manera posible.
2. Se requiere un lenguaje de programación llamado SQL para el acceso y manejo de las bases lo que nos proporciona integridad fiabilidad y seguridad de los datos.
3. Podemos asignar niveles de acceso a ciertas bases o incluso tablas lo que nos da una mayor seguridad en el manejo de los datos.
4. Muchos gestores de base de datos disponen de API’s visuales y muy fáciles de manejar para facilitar la gestión de los datos.

Para poder elegir un tipo de base de datos relacional o no relacional podemos observar en la siguiente imagen una comparación entre ellas y sus características.

BASE DE DATOS SQL	BASE DE DATOS NoSQL
Está basada en tablas	Está basada en documentos
Mantiene una estructura definida	Mantiene una estructura dinámica
Se diseñan para tener la menos redundancia	Permite la redundancia de datos
Necesitan más recursos	No necesita tantos recursos para ejecutarse
Entre más compleja mayor procesamiento requiere	Necesita menos procesamiento de datos y más de archivos.

Tabla 3. TABLA COMPARATIVA SQL VS NOSQL

SQL vs NoSQL		
Características	BASE DE DATOS SQL	BASE DE DATOS SQL
Ventajas	•Estándares bien definidos •Estructura simple •Hay bastante información respecto a este tipo de bases	•Son muy versátiles •Hacer cambios sin necesidad de alterar las estructuras •No necesita servidores con gran cantidad de recursos
Desventajas	•Crecimiento en volumen lo que significa crecimiento de costos •Algunas bases SQL se ven afectas por el sistema operativo que utilicemos	•Hay menos información al respecto que las SQL •La mayoría de las bases de datos NoSQL no tienen una interfaz grafica
Comparativa	•Está basada en tablas •Se diseñan para tener la menos redundancia •Necesitan más recursos lo que significa que entre más compleja sea requerimos más procesamiento	•Está basada en documentos •Mantienen una estructura dinámica •Permite la redundancia de datos •No necesitas tantos recursos para ejecutarse

Tabla 4. TABLA COMPARATIVA VENTAJAS Y DESVENTAJAS DE SQL Y NOSQL

Dado el análisis anterior nuestra mejor opción debido a la cantidad de datos que se van a obtener a partir de las estaciones meteorológicas no afectan en rendimiento o recursos de memoria y es por ello que la mejor opción que se considera optima es utilizar bases de datos SQL para implementar dicho proyecto.

4.2.1 MANEJADORES DE BASE DE DATOS

Es un software específico para el manejo de las bases de datos y la principal función es ser una interfaz entre las bases de datos y el usuario.

Existen varios tipos de manejadores de base de datos, en la tabla anterior, se presenta una tabla comparativa de las diferencias entre los manejadores más populares de base de datos.

	Manejador de Base de Datos				
Características	**MySQL**	**Microsoft SQL server**	**PostgreSQL**	**Oracle Database**	**SQLite**
Ventajas	Manejo sencillo Código abierto Fácil configuración Gran velocidad a la hora de realizar operaciones	Gran interfaz gráfica Seguridad	Gran capacidad de almacenamiento	Gestión de múltiples bases de datos Ofrece soporte técnico	Podemos utilizar varios lenguajes de programación Pequeño tamaño
Desventajas	No soporta algunas conversiones de datos	Utiliza muchos recursos para un procesamiento adecuado	Lento en inserciones y actualizaciones	configuración compleja	Esta limitado debido a la cantidad de especio necesita
Sistemas operativos	FreeBSD, HP-UX, GNU/Linux, Mac OS X, Warp, QNX, SGI IRIX, Solaris, SunOS,	Windows, Linux	OpenBSD, GNU/Linux, Mac OS, Windows 95	Linux, OS X, Solaris, Windows/OS	Windows CE & Pocket PC, Mac OSX, Linux

	Tru64, Windows 95				

Tabla 5. TABLA COMPARATIVA ENTRE MANEJADORES DE BASES DE DATOS

4.3 ANALISIS DE ALGORITMOS DE MACHINE LEARNING

Y referente a los manejadores de bases de datos es importante considerar la tecnología que se va a implementar para la computación en la nube es AWS es MySQL por lo que es el manejador que seleccionamos para le desarrollo de esta implementación.

Después del agrupamiento en clústeres, se asigna un número llamado ID de clúster a cada clúster. Ahora puedes condensar todo el conjunto de atributos de un ejemplo en su ID de clúster. Representar un ejemplo complejo con un ID de clúster simple hace que la agrupación en clústeres sea potente. Si extiendes la idea, los datos del agrupamiento en clústeres pueden simplificar grandes conjuntos de datos. Los sistemas de aprendizaje automático pueden usar los ID de clúster para simplificar el procesamiento de grandes conjuntos de datos. Por lo tanto, el resultado del agrupamiento en clústeres sirve como datos de atributos para sistemas de AA descendentes.

El objetivo del algoritmo SVM es encontrar un hiperplano que separe de la mejor forma posible dos clases diferentes de puntos de datos. “De la mejor forma posible” implica el hiperplano con el margen más amplio entre las dos clases, representado por los signos más y menos en la siguiente figura. El margen se define como la anchura máxima de la región paralela al hiperplano que no tiene puntos de datos interiores. El algoritmo solo puede encontrar este hiperplano en problemas que permiten separación lineal; en la mayoría de los problemas prácticos, el algoritmo maximiza el margen flexible permitiendo un pequeño número de clasificaciones erróneas. [29]

Las aplicaciones que utilizan reconocimiento de objetos y visión artificial, tales como las aplicaciones para vehículos autónomos y para reconocimiento facial, dependen en gran medida de CNN. [27]

Los modelos de regresión lineal son relativamente simples y proporcionan una fórmula matemática fácil de interpretar para generar predicciones. La regresión lineal es una técnica estadística establecida y se aplica fácilmente al software y a la computación. Las empresas lo utilizan para convertir datos sin procesar de manera confiable y predecible en inteligencia empresarial y conocimiento práctico. Los científicos de muchos campos, incluidas la biología y las ciencias del comportamiento, ambientales y sociales, utilizan la regresión lineal para realizar

análisis de datos preliminares y predecir tendencias futuras. Muchos métodos de ciencia de datos, como el Machine Learning y la inteligencia artificial, utilizan la regresión lineal para resolver problemas complejos. [26]

Los datos que se recopilarán desde las estaciones meteorológicas serán:

- Temperatura y humedad.
- Velocidad y dirección del viento.
- Lluvia actual y acumulada diaria, mensual y anual.
- Intensidad de lluvia.
- Presión atmosférica actual y tendencia.

Considerando las variables anteriores, se deben recabar datos que serán depositados en una base de datos que permita almacenar dichos datos de manera organizada.

Se integrarán datos de la red de calidad del aire de la CDMX así mismo se integrará otro dataset del Observatorio Hidrológico de la UNAM, POWER Data Access Viewer de la NASA, Comisión Nacional del Agua, así como del centro de Investigación Científica y de Educación Superior de Ensenada (CICESE) los cuales tienen datos de eventos pluviales.

La siguiente tabla contiene los parámetros que se generan a partir de las estaciones meteorológicas, así como los datos que podemos consultar a través de los dataset disponibles para probar nuestro modelo de Machine Learning.

COMPARACION DE INFORMACION DE PORTALES ADICIONALES												
	Temp	**Velocidad del viento**	**Direc. de viento**	**Lluvia actual**	**Lluvia diaria**	**Lluvia mensual**	**Lluvia anual**	**Presión At.**	**Radiación e índice UV**	**Fase lunar**	**Puesta y salida del sol**	**Hum**
Estación Meteorológica UPIITA	✓	✓	✓	✓	✓	✗	✗	✓	✗	✓	✗	✓
Estación Meteorológica ESCOM	✓	✓	✓	✓	✓	✗	✗	✓	✗	✓	✗	✓
Red de la calidad del aire de CDMX	✓	✓	✓	✗	✗	✗	✗	✓	✓	✗	✗	✓
Observatorio Hidrológico de la UNAM	✗	✗	✗	✓	✓	✓	✗	✗	✗	✗	✗	✓
POWER Data Access Viewer	✓	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗	✓
Comisión Nacional del Agua	✗	✓	✗	✓	✓	✓	✓	✗	✗	✗	✗	✓
Centro de Investigación Científica y de Educación Superior de Ensenada	✓	✗	✗	✗	✓	✓	✓	✗	✗	✗	✗	✗

Tabla 6. TABLA COMPARATIVA ENTRE PARÁMETROS METEOROLÓGICO

4.4. ARQUITECTURAS MONOLITICAS Y DE MICROSERVICIOS

En el desarrollo de software existen diferentes tipos de arquitectura para la creación de un sistema, dicha arquitectura ayuda nos ayuda a poder llevar un control entre el FrontEnd, BackEnd y el modelo de datos, nosotros debido a la dificultad que conlleva el Proyecto Terminal, debemos basarnos en arquitectura de microservicios o monolítica.

En la siguiente tabla podemos ver una comparación entre dichas arquitecturas:

Arquitecturas		
	Microservicios	Monolitica
Definición	Es un método de arquitectura que se basa en una serie de servicios que se pueden implementar de forma independiente. Estos servicios tienen su propia lógica empresarial y base de datos con un objetivo específico.	Es un modelo tradicional de un programa de software que se compila como una unidad unificada y que es autónoma e independiente de otras aplicaciones.
Ventajas	Agilidad Escalado flexible Implementación Continua Facil mantenimiento y pruebas Implementación independiente Flexibilidad tecnológica Alta fiabilidad	Implementación Sencilla Pruebas Simplificadas Depuración Sencilla
Desventajas	Costes exponenciales de infraestructura Desafíos para la depuración Falta de estandarización Desarrollo descontrolado Sobrecarga Organizativa	Velocidad de desarrollo más lenta Escalabilidad Falta de flexibilidad Implementación Barrera para la adopción de tecnología

Tabla 7. TABLA COMPARATRIVA DE ARQUITECTURA MONOLITICA VS MICROSERVICIOS

Teniendo en cuenta la tabla anterior, lo que mejor se adapta a nuestro sistema es la arquitectura de microservicios ya que tiene bastantes ventajas sobre la arquitectura monolítica.

4.4.1 COMPARACIÓN ENTRE LOS DIFERENTES LENGUAJES DE PROGRAMACIÓN

Características	Lenguajes de Programación BackEnd			
	C	C++	Java	Python
Descripción	C es un programa estructurado de alto y bajo nivel simultaneamente.	C++ es un lenguaje compilado que fue desarrollado como una extensión orientada a objetos del lenguaje C.	Java es un lenguaje de programación de alto nivel, basado en clases y orientado a objetos	Python es un lenguaje de programación de alto nivel, se trata de un lenguaje interpretado.
Rendimiento	Este lenguaje tiene estructuras de datos simples con traducción directa al lenguaje máquina, provocando un excelente performance.	El rendimiento que ofrece es alto. Esto es debido a que puede hacer llamadas directas al sistema operativo, posee gran variedad de parámetros de optimización y se integra de forma directa con el lenguaje ensamblador.	La mayoría de las JVM (máquina virtual de Java) realizan una compilación «just-in-time» de todo o parte de los programas a código nativo, lo que mejora significativamente el rendimiento.	Python es ampliamente reconocido como lento. Aunque Python nunca alcanzará el rendimiento de lenguajes de bajo nivel como C, Fortran o incluso Java
Ventajas	Economía en las expresiones Abundancia de operadores y tipos de datos Producción de código objeto altamente optimizado. No está orientado a un area en especial.	Alto rendimiento Lenguaje Actualizado Multiplataforma Extendido	Es un lenguaje de código abierto Tiene una compilación de códigos seguro Admite subprocesos múltiples Es económico Es seguro Brinda la posibilidad de crear aplicaciones distribuidas Ofrece una gestión de error Es compatible con librerías estándar y editores Cuenta con liberación de memoria Es multiplataforma Está orientado a los objetos	Lenguaje de alto nivel Polivalente y de paradigmas Bibliotecas y frameworks Portabilidad Gratis y de código abierto

Tabla 8. TABLA COMPARATIVA ENTRE LENGUAJES DE PROGRAMACIÓN

En la trayectoria escolar se enseña y desarrolla en distintos lenguajes de programacion tales como C, C++, Python, JAVA, etc. Debido a esto y las ventajas que tienen los lenguajes programación JAVA y Python sobre los demás lenguajes de programación estos fueron lo que se seleccionaron; JAVA para la parte de la arquitectura de microservicios, la lógica de negocio y Python para el desarrollo de nuestro modelo de Machine Learning.

Tabla 9. TABLA COMPARATRIVA DE ARQUITECTURA MONOLITICA VS MICROSERVICIOS [25]

4.5. MARCOS DE TRABAJO

Dados los lenguajes de programación seleccionados se hizo una comparación entre marcos de trabajo para poder implementar el Proyecto Terminal.

Marco de trabajo	Laravel	Flask	Spring Boot	ASP. Net Core
Tipo	Marco de trabajo	Micro Marco de trabajo	Marco de trabajo	Marco de trabajo
Disponibilidad de recursos de aprendizaje	1,784	645	2,729	3,117
Comunidad	Tiene una amplia comunidad debido a que PHP domina el 79% del mercado en lo que respecta a la programación del lado de servidor.	Su comunidad no es tan amplia debido a que es un marco relativamente nuevo.	Es el marco más popular de Java, debido a su velocidad, simplicidad y productividad. Respaldada por comunidad de código abierto	Amplia comunidad debido a su versatilidad para realizar aplicaciones web. Respaldada por Microsoft.
Rendimiento	Ofrece buen rendimiento debido a que aprovecha las virtudes de la memoria cache.	Fue diseñado para proveer alto rendimiento para la mayoría de los usuarios, debido a que tiene pocas capas de abstracción.	Su rendimiento es regular en comparación a otros marcos, sus virtudes no se centran en el rendimiento.	Su mayor ventaja respecto a otros marcos es el rendimiento, debido a que recibe continuamente actualizaciones.
Documentación	Amplia documentación y comunidad debido a su dominio del mercado.	Gracias a su sencillez, la documentación oficial es suficiente para empezar a desarrollar aplicaciones.	Amplia documentación y comunidad debido a su amplio uso en la industria.	Cuenta con la documentación más amplia y precisa, pues esta es provista por Microsoft.
Curva de aprendizaje	Curva de aprendizaje regular gracias a que PHP es un lenguaje sencillo y no requiere conocimientos previos.	Tiene la curva de aprendizaje menos pronunciada, debido a que es un marco minimalista y su orientación es al desarrollo acelerado.	Curva de aprendizaje más pronunciada en comparación de los otros marcos, debido su filosofía de convención sobre configuración.	Curva de aprendizaje menos pronunciada debido a que no requiere tanto código.
Mantenibilidad	Desarrollo orientado a módulos.	Desarrollo orientado a módulos.	Desarrollo orientado a Módulos.	Desarrollo orientado a Módulos.
Lenguaje Base	PHP	Python	Java	C#

Tabla 10. TABLA COMPARATIVA DE MARCOS DE TRABAJO

4.5.1 MARCOS DE TRABAJO PARA FRONT.

Marco de trabajo	Angular	Vue	React	Ember
Tipo	Marco de trabajo	Marco de trabajo	Librería	Marco de trabajo
Disponibilidad de recursos de aprendizaje	2,982	784	4,873	26
Comunidad	Al ser el marco más establecido, por tanto, su comunidad es enorme. Respaldada por Google.	Su comunidad ha aumentado, sin embargo, no es un marco tan establecido como Angular. Respaldada por fundaciones de código abierto.	Es la librería con la mayor comunidad. Respaldada por Facebook	Nunca tuvo una comunidad considerable, debido a la inmensa curva de aprendizaje. Respaldada por fundaciones de código abierto
Rendimiento	No es tan rápido como React, pero es debido a que cuenta con un enlace de datos bidireccional	Comparte con Angular el DOM virtual, los componentes reactivos y componibles lo hacen tan rápido como Angular.	Es la opción más rápida disponible debido a que es sólo una biblioteca orientada al DOM y el enlace unidireccional.	Es la opción con el menor rendimiento, especialmente renderizando la página por primera vez.
Documentación	Una de las mejores documentaciones, debido a que es simple, corta y al punto.	Debido a su sencillez, la documentación oficial es suficiente para empezar a trabajar.	Debido a su flexibilidad, es difícil encontrar la documentación exacta puede resultar complicado.	Aunque la documentación oficial es regular, debido a su poca comunidad es difícil encontrar información en foros o blogs.
Curva de aprendizaje	Su curva es más pronunciada en comparación a otros marcos, debido a que la documentación es extremadamente grande y requiere familiarización con algunos conceptos antes de trabajar.	Su curva es la menos pronunciada, debido a que sólo se necesita conceptos básicos de HTML, CSS y JS. Se puede crear aplicaciones simples y no triviales en menos de un día.	Su curva no es tan pronunciada gracias a que es una biblioteca básica con pocas API y flujos de trabajo. Se puede utilizar con mayor facilidad.	Su curva es más pronunciada incluso que Angular, debido a que convención sobre el concepto de configuración, por lo que familiarizarse con la convención llevará tiempo.
Mantenibilidad	Desarrollo orientado a componentes	Desarrollo orientado a componentes	Desarrollo orientado a componentes	Desarrollo orientado a componentes

Lenguaje Base	TypeScript	JavaScript	JavaScript	JavaScript

Tabla 11. TABLA COMPARATIVA DE MARCOS DE TRABAJO PARA FRONT

4.6 ANALISIS DE ALGORITMOS DE MACHINE LEARNING

En su forma más simple, la inteligencia artificial es un campo que combina la ciencia informática y los conjuntos de datos robustos para permitir la resolución de problemas, que pueden ser abarcados con los subcampos de “Machine Learning” y “Deep learning” ambas diciplinas están compuestas por algoritmos de IA que buscan crear sistemas expertos que hagan predicciones o clasificaciones basadas en datos de entrada.

De manera básica el aprendizaje automático se basa en el uso de algoritmos informáticos programados que reciben y analizan conjuntos de datos de entrada para predecir datos de salida dentro de un rango aceptable.

Existen tres tipos de algoritmos de aprendizaje automático que existen: supervisado, no supervisado y por refuerzo.

Aprendizaje supervisado

Se produce cuando el operador introduce en el algoritmo un conjunto de datos con entradas y salidas, de esa manera el algoritmo debe de encontrar métodos para llegar a las salidas dadas, donde el algoritmo reconoce patrones, aprende de las observaciones y hace predicciones.

Aprendizaje no supervisado

Se produce cuando el algoritmo estudia los datos para identificar patrones, pero no hay calve de respuesta ni ningún operador humano para dar instrucciones, la máquina analiza las relaciones y correlaciones con los análisis de datos que tiene disponibles.

Aprendizaje por esfuerzo

Se centra en los procesos de aprendizajes reglamentados con un conjunto de acciones, parámetros y valores finales, al definir las reglas, el algoritmo de aprendizaje automático intenta explorar diferentes posibilidades, evaluando cada resultado para determinar cuál es el óptimo.

Este sistema enseña la máquina a través de ensayo y error.

A continuación, se presentan algunos tipos de algoritmos de “Machine Learning” y sus definiciones.

- Algoritmos de regresión lineal

Muestran o predicen la relación entre dos variables o factores poniendo una línea recta continua a los datos. La regresión lineal es uno de los tipos de análisis de regresión más populares.

- Algoritmos de regresión logística

  Ponen una curva continua en forma de S en los datos. La regresión logística es otro tipo popular de análisis de regresión.

- AlgoritmOS Naïve Bayes

  Calculan la probabilidad de que se produzca un evento en función de que ocurra un evento relacionado.

- Algoritmos de agrupación

  Sirven para categorizar datos no etiquetados, es decir, datos sin categorías o grupos definidos.

- Algoritmos de árbol decisión

  Dividen los datos en dos o más conjuntos homogéneos. Usan reglas if-then para separar los datos según el diferenciador más significativo entre los puntos de datos.

- K vecinos más cercanos

  Almacenan todos los puntos de datos disponibles y clasifican cada punto de datos nuevo en función de los puntos de datos más cercanos a él, medidos por una función de distancia.

- K-Means

  Clasifican los datos en clústeres, donde K es igual al número de clústeres. Los puntos de datos que están dentro de cada clúster son homogéneos y son heterogéneos respecto a los puntos de datos de otros clústeres.

Cabe destacar que al tener una gran cantidad de algoritmos tenemos que analizar los siguientes factores para poder elegir el algoritmo correcto.

- Parametrización: permite controlar la calidad del modelo resultante. Se pueden clasificar como:

  - Sin parámetros
  - Débiles
  - Simples / intuitivos
  - No intuitivo

- Tamaño de memoria: es el espacio de almacenamiento que necesitamos para guardar datos y variables, de manera que las clasificamos como:

- Pequeño
- Grande
- Extragrande

- Tendencia al sobre flujo: aparece cuando un modelo intenta predecir una tendencia en datos que es demasiado ruidosa, de manera que el sobreajuste es el resultado de un modelo demasiado complejo con demasiados parámetros lo cual ocasiona que sea inexacto ya que su tendencia no refleja la realidad de los datos.
  Existen muchas técnicas que se pueden usar para mitigar el sobreajuste tales como:

  La validación cruzada, la regularización, la detención temprana, la poda, los antecedentes bayesianos, el abandono y la comparación de modelos. La tendencia al sobre flujo se clasifica como:

  - Baja
  - Promedio
  - Alta
  - Muy alta

- Tiempo de entrenamiento: es el tiempo asociado con el entrenamiento del conjunto de datos y varia con respecto al tamaño de este y al algoritmo que estamos usando. Se clasifica como:

  - Débil
  - Costoso
  - Muy costoso

- Tiempo de predicción: es el tiempo asociado con la prueba del conjunto de datos, de manera que varía con el tamaño del conjunto de datos. Se clasifica como:

  - Débil
  - Costoso

4.7. TABLA COMPARATIVA DE ALGORIMOS DE MACHINE LEARNING

	Algoritmos de Machine Learning						
	Regresión lineal	Regresión logística	Naïve Bayes	Agrupación	Árbol de decisión	K vecinos más cercanos	K-Means
Descripción	Es una variación del algoritmo de árboles de decisión que ayuda a calcular una relación lineal entre una variable dependiente e independiente	Proviene de la estadística y a pesar de su nombre no es algoritmo para aplicar en regresión si no en clasificación	Son algoritmos de aprendizaje automático simples pero potentes. Se basan en la probabilidad condicional y el teorema de Bayes.	La agrupación es un poderoso método de aprendizaje de máquinas que implica la agrupación de puntos de datos.	Algoritmo de aprendizaje supervisado no paramétrico, tiene una estructura de árbol jerárquica, que consta de un nodo raíz, ramas, nodos internos y nodos hoja.	Es uno de los algoritmos de clasificación más básicos y esenciales en Machine Learning. Pertenece al dominio del aprendizaje supervisado	Algoritmo de clasificación no supervisada (clusterización) que agrupa objetos en k grupos basándose en sus características.
Uso	Usa las relaciones para hacer predicciones.	Su principal aplicación los problemas de clasificación binaria.	Clasificación basada en el Teorema de Bayes con una suposición de independencia entre los predictores	Categorizar o clasificar cada punto de datos en un grupo particular.	Se utiliza tanto para tareas de clasificación como de regresión.	Reconocimiento de patrones, minería de datos y detección de intrusos.	Clasificación de grupos.
Ventajas	•Admite la obtención de detalles. •Admite el uso de modelos de minería de datos OLAP.	•Eficacia y simplicidad. •No son necesarios grandes recursos computacionales	•Eficacia al analizar gran cantidad de datos	•Fácil de entender y aplicar	•Flexible. •Fácil de interpretar. •No se requiere preparación de datos.	•No paramétrico. •Algoritmo simple. •Alta precisión. •Insensible a datos atípicos.	•Facil de entender e implementar en código.
Desventajas	•No se admite el uso del Lenguaje predicción para crear modelos de minería de datos. •No admite la creación de dimensiones de minería de datos.	•Imposibilidad de resolver directamente problemas no lineales.	•Asume predictores independientes. En la vida real, es casi imposible que obtengamos un conjunto de predictores que sean completamente independientes.	•No es conveniente para grandes cantidades de datos. •Requiere que el usuario defina el número de clústeres.	•Propenso al sobreajuste. •Más costoso.	•Computacionalmente costoso. •Requisitos de memoria alta.	•Seleccionar cuantos clústeres hay no es nada trivial. •Puede producir distintos resultados en distintas ejecuciones.
Datos requeridos	Variables dependientes (Datos de entrada)	Valores de entrada (factores de análisis)	Datos de entrada, variables independientes.	Datos de entrada, definir el número de clústeres.	Datos de entrada	Datos de entrada.	Datos de entrada.

Tabla 12. TABLA COMPARATIVA ALGORITMOS DE MACHINE LEARNING

Clustering (Identificar patrones):

En el aprendizaje automático, a menudo también agrupamos ejemplos como primer paso para comprender un tema (conjunto de datos) en un sistema de aprendizaje automático. La agrupación de ejemplos sin etiqueta se denomina agrupamiento en clústeres. [28]

Como los ejemplos no tienen etiquetas, el agrupamiento en clústeres se basa en el aprendizaje automático no supervisado. Si los ejemplos están etiquetados, el agrupamiento en clústeres se convierte en clasificación. Para obtener un análisis más detallado de los métodos supervisados y no supervisados, consulta Introducción al enmarcado de problemas de aprendizaje automático.

El agrupamiento en clústeres tiene innumerables usos en una variedad de industrias. Algunas aplicaciones comunes para el agrupamiento en clústeres son las siguientes:

- segmentación de mercado
- análisis de redes sociales
- agrupación de resultados de la búsqueda
- diagnóstico por imágenes
- segmentación de imágenes
- detección de anomalías

Después del agrupamiento en clústeres, se asigna un número llamado ID de clúster a cada clúster. Ahora puedes condensar todo el conjunto de atributos de un ejemplo en su ID de clúster. Representar un ejemplo complejo con un ID de clúster simple hace que la agrupación en clústeres sea potente. Si extiendes la idea, los datos del agrupamiento en clústeres pueden simplificar grandes conjuntos de datos. Los sistemas de aprendizaje automático pueden usar los ID de clúster para simplificar el procesamiento de grandes conjuntos de datos. Por lo tanto, el resultado del agrupamiento en clústeres sirve como datos de atributos para sistemas de AA descendentes.

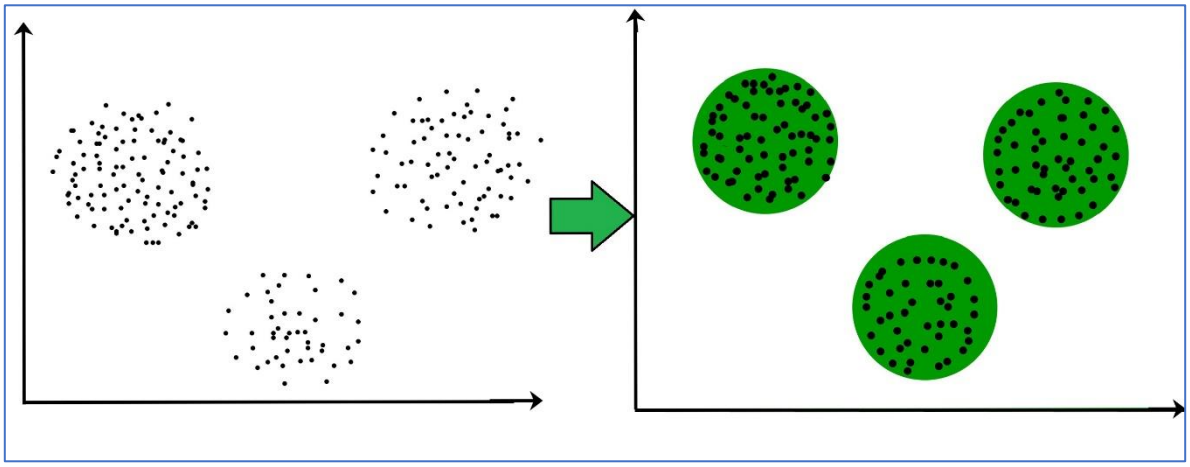


Ilustración 18. Algoritmo Machine Learning Clustering

Support Vector Machine:

Support vector machine (SVM) es un algoritmo de aprendizaje supervisado que se utiliza en muchos problemas de clasificación y regresión, incluidas aplicaciones médicas de procesamiento de señales, procesamiento del lenguaje natural y reconocimiento de imágenes y voz.

El objetivo del algoritmo SVM es encontrar un hiperplano que separe de la mejor forma posible dos clases diferentes de puntos de datos. “De la mejor forma posible” implica el hiperplano con el margen más amplio entre las dos clases, representado por los signos más y menos en la siguiente figura. El margen se define como la anchura máxima de la región paralela al hiperplano que no tiene puntos de datos interiores. El algoritmo solo puede encontrar este hiperplano en problemas que permiten separación lineal; en la mayoría de los problemas prácticos, el algoritmo maximiza el margen flexible permitiendo un pequeño número de clasificaciones erróneas.

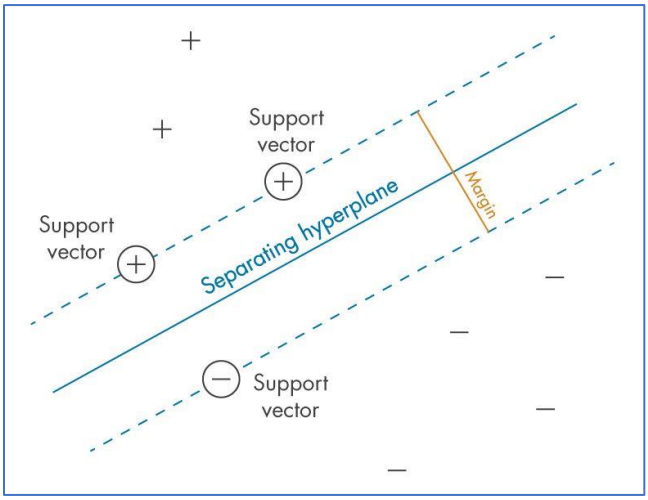


Ilustración 19. Algoritmo de Machine Learning Support Vector

Redes Neuronales Convolucionales:

Una red neuronal convolucional (CNN) es una arquitectura de red para deep learning que aprende directamente de los datos, sin necesidad de extraer características manualmente.

Estas redes son particularmente útiles para encontrar patrones en imágenes para reconocer objetos, caras y escenas. También resultan eficaces para clasificar datos sin imágenes, tales como datos de audio, series temporales y señales.

Las aplicaciones que utilizan reconocimiento de objetos y visión artificial, tales como las aplicaciones para vehículos autónomos y para reconocimiento facial, dependen en gran medida de CNN.

Las CNN proporcionan una arquitectura óptima para descubrir y aprender características principales en imágenes y datos de series temporales. Las CNN son una tecnología clave en aplicaciones tales como:

- Imágenes médicas: las CNN pueden examinar miles de informes patológicos para detectar visualmente la presencia o ausencia de células cancerosas en las imágenes.
- Procesamiento de audio: la detección de palabras clave se puede utilizar en cualquier dispositivo con un micrófono para detectar cuándo se pronuncia una palabra o frase determinada ("Oye Siri"). Las CNN pueden aprender y detectar con precisión la palabra clave e ignorar todas las demás frases, independientemente del entorno.
- Detección de señales de stop: la conducción autónoma se basa en CNN para detectar con precisión la presencia de una señal u otro objeto y tomar decisiones basadas en el resultado.
- Generación de datos sintéticos: utilizando redes generativas antagónicas (GAN), se pueden producir nuevas imágenes para su uso en aplicaciones de deep learning, tales como reconocimiento facial y conducción autónoma.

Regresión lineal (Pronósticos):

Es un algoritmo de machine learning que se utiliza para realizar predicciones. Una regresión lineal se utiliza para generar predicciones sobre una variable que llamamos la variable dependiente, dadas unas variables independientes, una regresión lineal en su forma más simple es una línea recta en dos dimensiones. que se ajusta a unos valores de datos.

La regresión lineal es el primer método para revisar y utilizar cuando se intente hacer predicciones sobre una variable numérica.

La regresión lineal es un método estadístico habitual que se han adoptado en Machine Learning y se ha mejorado con muchos métodos nuevos de ajuste de las líneas y medición de los errores. Simplemente, la regresión se refiere a la predicción de un objetivo numérico. La regresión lineal sigue siendo una buena opción cuando se desee un modelo simple para una tarea de predicción básica. La regresión lineal también suele funcionar bien en conjuntos de datos muy dimensionales, dispersos y sin complejidad.

Recordemos que los algoritmos de Machine Learning Supervisados, aprenden por sí mismos y -en este caso- a obtener automáticamente esa “recta” que buscamos con la tendencia de predicción. Para hacerlo se mide el error con respecto a los

puntos de entrada y el valor “Y” de salida real. El algoritmo deberá minimizar el coste de una función de error cuadrático y esos coeficientes corresponderán con la recta óptima. Hay diversos métodos para conseguir minimizar el coste. Lo más común es utilizar una versión vectorial y la llamada Ecuación Normal que nos dará un resultado directo.

Los modelos de regresión lineal son relativamente simples y proporcionan una fórmula matemática fácil de interpretar para generar predicciones. La regresión lineal es una técnica estadística establecida y se aplica fácilmente al software y a la computación. Las empresas lo utilizan para convertir datos sin procesar de manera confiable y predecible en inteligencia empresarial y conocimiento práctico. Los científicos de muchos campos, incluidas la biología y las ciencias del comportamiento, ambientales y sociales, utilizan la regresión lineal para realizar análisis de datos preliminares y predecir tendencias futuras. Muchos métodos de ciencia de datos, como el Machine Learning y la inteligencia artificial, utilizan la regresión lineal para resolver problemas complejos.

Como parte de nuestra investigación debido a que seleccionamos los servicios de ASW, dentro de sus herramientas de Machine Learning podemos encontrar:

Amazon SageMaker

Es un servicio de aprendizaje automático completamente administrado. Con SageMaker, se pueden crear y entrenar modelos de aprendizaje automático de forma rápida y sencilla, y luego implementarlos directamente en un entorno alojado listo para la producción. Proporciona una instancia de notebook de creación de Jupyter (Python) integrada para acceder fácilmente a las fuentes de datos para exploración y análisis, de modo que no se tenga que administrar servidores. También proporciona algoritmos comunes de aprendizaje automático que están optimizados para ejecutarse de manera eficiente con datos extremadamente grandes en un entorno distribuido.

Aprendizaje automático en AWS SageMaker

El aprendizaje automático es un proceso iterativo. Requiere herramientas de flujo de trabajo y hardware dedicado para procesar conjuntos de datos. En un escenario típico, un equipo de ciencia de datos construye modelos Machine Learning en dos pasos o canalizaciones: entrenamiento e inferencia.

El entrenamiento de datos le enseña a una máquina a comportarse de cierta manera en función del reconocimiento de patrones recurrentes dentro de los conjuntos de datos. Luego, los datos se infieren o entrenan para responder a nuevos patrones de datos. Una vez que los científicos de datos ajustan el modelo de Machine Learning, los equipos de desarrollo de software convierten el modelo terminado en interfaces de programas de aplicaciones (API) de productos o servicios.

Muchas empresas no tienen el presupuesto para traer especialistas y mantener recursos dedicados al desarrollo de IA. AWS SageMaker utiliza herramientas integradas para automatizar procesos manuales que requieren mucha mano de obra y reducir los errores humanos y los costos de hardware. Los componentes de modelado de Machine Learning se empaquetan en un conjunto de herramientas de AWS SageMaker. Las capacidades del software se resumen en plantillas intuitivas de SageMaker. Proporcionan un marco para construir, alojar, entrenar e implementar modelos Machine Learning a escala en la nube pública de Amazon.

4.8 DATASETS

DATOS DEL DATASET

- IdEstacion int
- Fecha timestamp
- Hora time
- Temperatura int
- VelocidadViento int
- DireccionViento varchar
- LluviaActual int
- LuviaDiaria int
- PresionAdmosferica int
- Humedad int

FRECUENCIA DE MUESTREO
toma de muestras de cada uno de los parámetros cada hora.

ID_registro	ID estacion	Fecha	Hora	Temp	Velocidad del viento	Direc. de viento	Lluvia actual	Lluvia diaria	Presión At.	Hum
int	int	Data	Time	int	int	String/varchar	int	int	int	int

Dataset completo generado por la estación meteorológica DAVIS VANTAGE PRO2 (estaciones meteorológicas UPIITA/ESCOM)

- Fecha
- Hora
- Temperatura
- Humedad
- Dirección del viento

- Velocidad del viento
- Lluvia
- Ritmo de Lluvia
- Radiación solar
- Índice UV
- Densidad del aire

Dataset completo generado por la estación meteorológica dirección de calidad del aire CDMX

- Ozono
- Dióxido de nitrógeno
- Óxido nítrico
- Óxidos de nitrógeno
- Monóxido de carbono
- Dióxido de azufre
- Partículas menores a 10 micrómetros
- Partículas menores a 2.5 micrómetros
- Partículas coarse
- Benceno
- Tolueno
- Velocidad del viento
- Dirección del viento
- Temperatura ambiente
- Humedad relativa
- Radiación UV-A
- Radiación UV-B
- Radiación global
- Radiación FA
- Presión Atmosférica
- Potencial de hidrógeno
- Precipitación pluvial
- Conductividad eléctrica
- Sulfato
- Nitrato
- Cloruro
- Carbonato
- Hidrógeno
- Amonio
- Calcio
- Magnesio
- Sodio
- Potasio

- Partículas suspendidas totales
- Plomo contenido en PST
- Plomo contenido en PM10
- Etilbenceno
- Xilenos

Dataset completo generado por el observatorio hidrológico UNAM

- Intensidad de precipitación
- precipitación acumulada
- Numero de gotas

Dataset enriquecido por las distintas fuentes de información

- Fecha
- Hora
- Temperatura
- Velocidad del viento
- dirección del Viento
- Lluvia Actual
- Luvia Diaria
- Presión atmosférica
- Humedad

Id_Estacion	Fecha	Hora	Temperatura	Velocidad del viento	Dir. del Viento	Lluvia Actual	Luvia Diaria	Presión atmosférica	Humedad
1	01/01/2022	04:05	11.19	3.48	195.75	0	0	74.36	4.64
2	01/01/2022	04:05	11.44	1.28	204.75	0	0	74.6	3.66
3	01/01/2022	04:05	11.84	2.65	209.12	0	0	74.67	4.76
4	01/01/2022	04:05	12.57	1.18	199.88	0	0	74.65	3.85
1	02/01/2022	05:05	10.44	1.33	171.75	0	0	74.57	5.43
2	02/01/2022	05:05	11.98	2.28	254.25	0	0	74.53	6.29
3	02/01/2022	05:05	12.99	1.6	160.94	0	0	74.62	6.77
4	02/01/2022	05:05	12.7	1.01	220.06	0	0	74.67	6.53

ID estacion	Nombre_Estacion	Ubicacion
int	String/ varchar	String/varchar
1	UPIITA	Ticoman CDMX
2	ESCOM	Zacatenco CDMX

4.9. PRUEBAS DE CONCEPTO ESTACIÓN METEOROLÓGICA

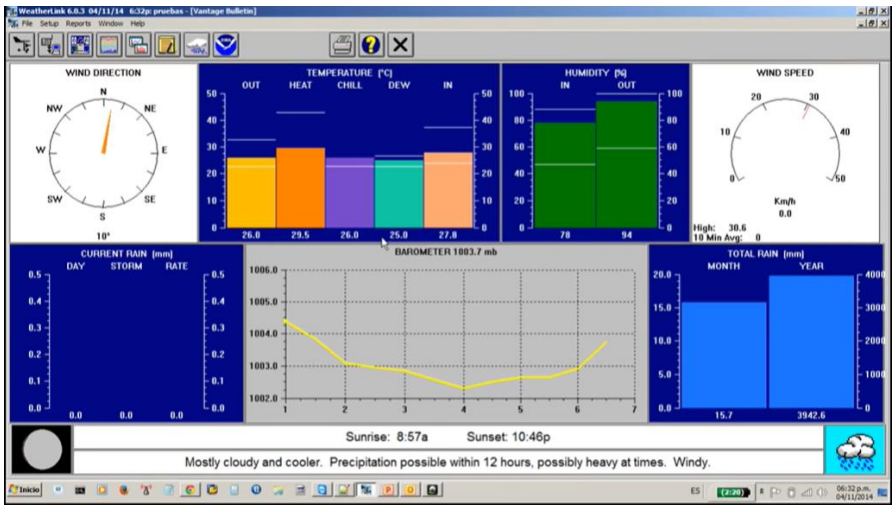
Para la configuración de estas dos estaciones meteorológicas se necesitó dividirla en varias etapas en las cuales se fueron probando varios aspectos con diferentes grados de dificultad.

1.-Se comenzó a armar la segunda estación meteorológica pues la primera ya está instalada en UPIITA, esta segunda estación es la que estará instalada en ESCOM. Para el proceso se requirió algunas horas de armado.

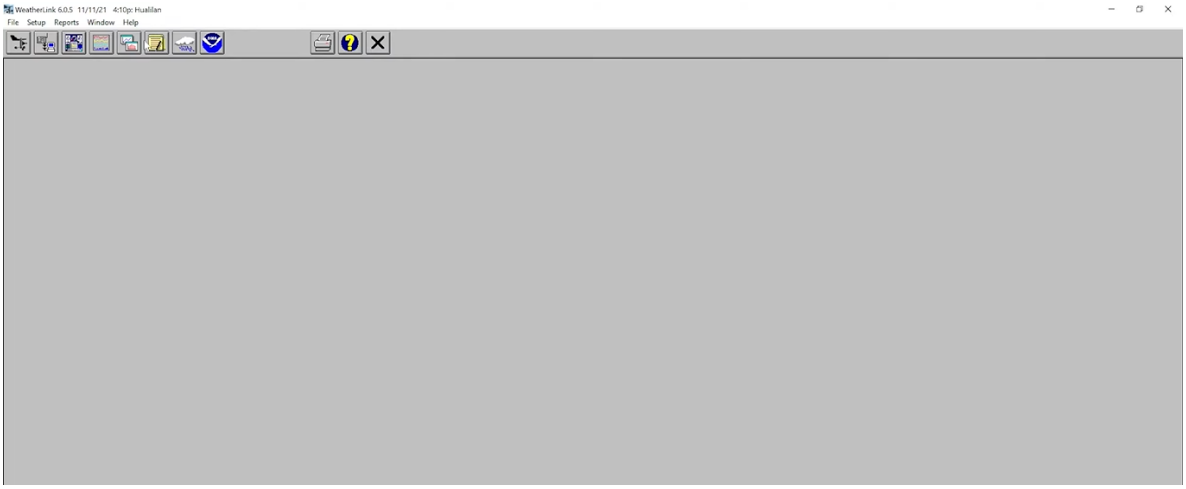
2.-Se calibraron ambas consolas para que estas comenzaran a recibir los datos de las estaciones.

En un inicio ambas consolas tomaban datos de una sola estación por lo cual el primer problema fue que cada consola tomara los datos de su respectiva estación, para esto cada estación tiene un dip switch en la parte del circuito, la combinación de orden de estos switches y una correcta configuración en la consola hace posible la conectividad con éxito.

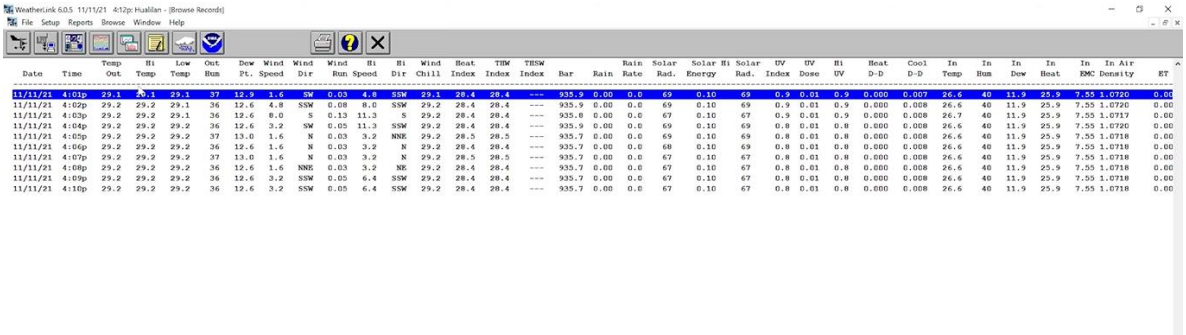
3.- Para la instalación del software (Weatherlink) se ingresó a la página oficial y se descargó el programa para la visualización de los datos en la PC la cual se puede ver de la siguiente manera:



Para esto tiene varias funciones el software donde se observa de manera gráfica y en forma de tabla, también da la opción de descargar los datos.

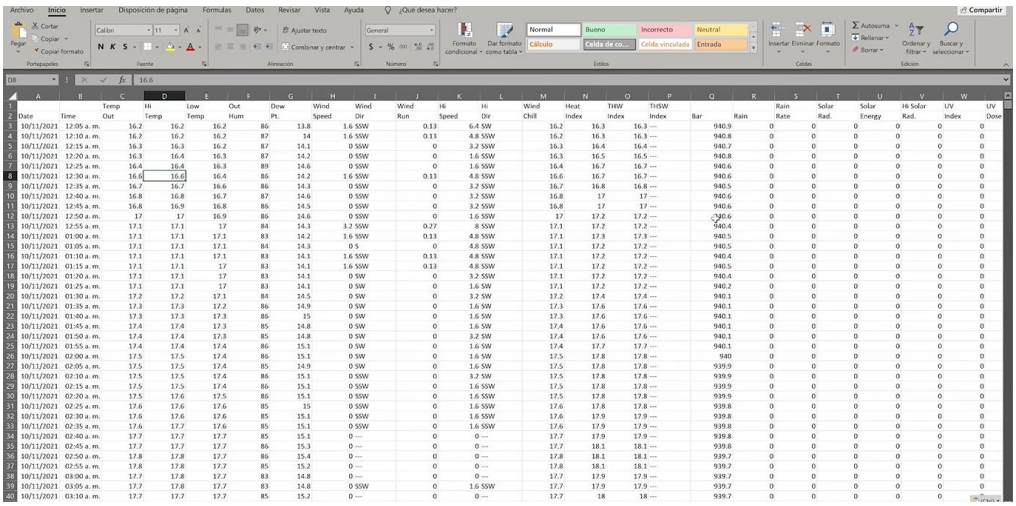


4.- Para la parte de guardado de datos la aplicación automáticamente va tomando los datos y los va guardando de manera local en la aplicación y se pueden visualizar de la siguiente forma:

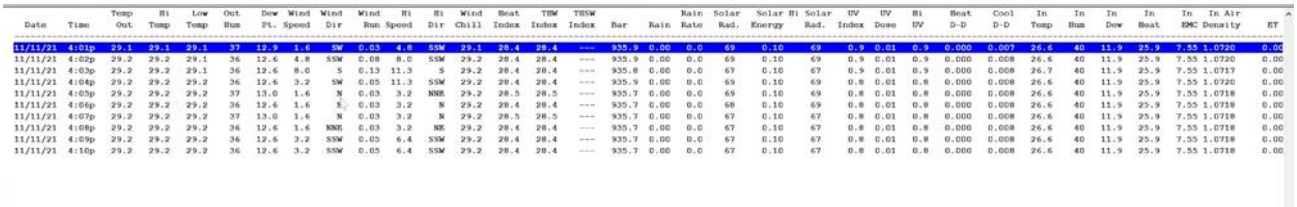


Date	Time	Temp Out	Hi Temp	Low Temp	Out Hum	Dew Pt.	Wind Speed	Wind Dir	Wind Run	Hi Speed	Hi Dir	Wind Chill	Heat Index	THW Index	THSW Index	Bar	Rain	Rain Rate	Solar Rad.	Solar Energy	Hi Solar Rad.	UV Index	UV Dose	Hi UV	Heat D-D	Cool D-D	In Temp	In Hum	In Dew	In Heat	In EMC	In Air Density	ET
11/11/21	4:01p	29.1	29.1	29.1	37	12.9	1.6	SW	0.03	4.8	SSW	29.1	28.4	28.4	---	935.9	0.00	0.0	69	0.10	69	0.9	0.01	0.9	0.000	0.007	26.6	40	11.9	25.9	7.55	1.0720	0.00
11/11/21	4:02p	29.2	29.2	29.1	36	12.6	4.8	SSW	0.08	8.0	SSW	29.2	28.4	28.4	---	935.9	0.00	0.0	69	0.10	69	0.9	0.01	0.9	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0720	0.00
11/11/21	4:03p	29.2	29.2	29.1	36	12.6	8.0	S	0.13	11.3	S	29.2	28.4	28.4	---	935.8	0.00	0.0	67	0.10	67	0.9	0.01	0.9	0.000	0.008	26.7	40	11.9	25.9	7.55	1.0717	0.00
11/11/21	4:04p	29.2	29.2	29.2	36	12.6	3.2	SW	0.05	11.3	SSW	29.2	28.4	28.4	---	935.9	0.00	0.0	69	0.10	69	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0720	0.00
11/11/21	4:05p	29.2	29.2	29.2	37	13.0	1.6	N	0.03	3.2	NNE	29.2	28.5	28.5	---	935.7	0.00	0.0	69	0.10	69	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0718	0.00
11/11/21	4:06p	29.2	29.2	29.2	36	12.6	1.6	N	0.03	3.2	N	29.2	28.4	28.4	---	935.7	0.00	0.0	68	0.10	69	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0718	0.00
11/11/21	4:07p	29.2	29.2	29.2	37	13.0	1.6	N	0.03	3.2	N	29.2	28.5	28.5	---	935.7	0.00	0.0	67	0.10	67	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0718	0.00
11/11/21	4:08p	29.2	29.2	29.2	36	12.6	1.6	NNE	0.03	3.2	NE	29.2	28.4	28.4	---	935.7	0.00	0.0	67	0.10	67	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0718	0.00
11/11/21	4:09p	29.2	29.2	29.2	36	12.6	3.2	SSW	0.05	6.4	SSW	29.2	28.4	28.4	---	935.7	0.00	0.0	67	0.10	67	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0718	0.00
11/11/21	4:10p	29.2	29.2	29.2	36	12.6	3.2	SSW	0.05	6.4	SSW	29.2	28.4	28.4	---	935.7	0.00	0.0	67	0.10	67	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0718	0.00

5.-Para la descarga de datos la aplicación ofrece dos opciones, la primera es descargar los datos en un archivo Excel:



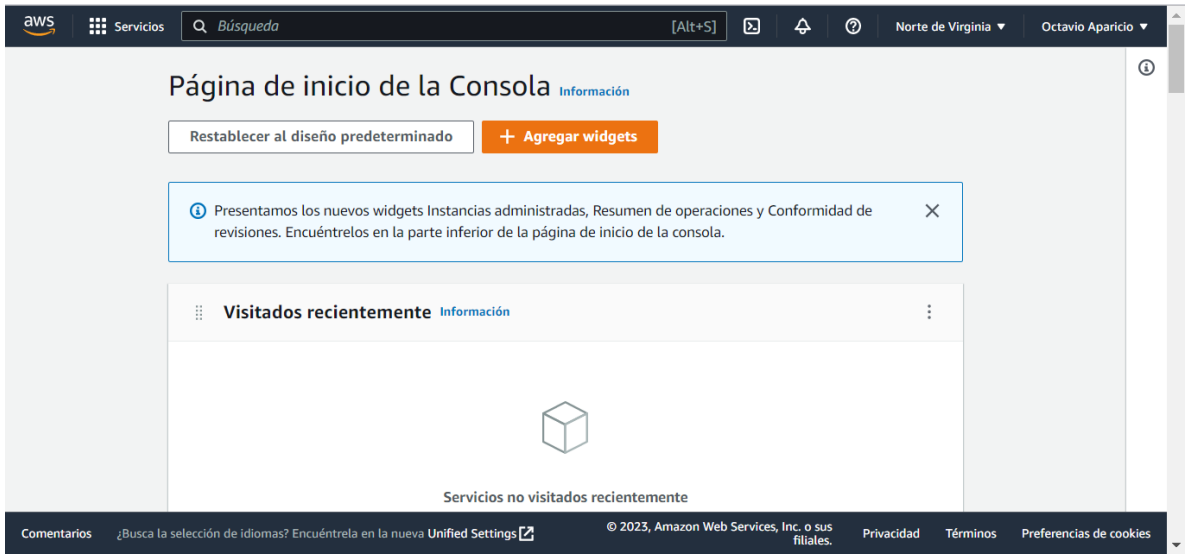
también da la opción de descargar los datos en formato txt en el mismo orden como se visualiza desde la aplicación:



Date	Time	Temp Out	Hi Temp	Low Temp	Out Hum	Dew Pt.	Wind Speed	Wind Dir	Wind Run	Hi Speed	Hi Dir	Wind Chill	Heat Index	THW Index	THSW Index	Bar	Rain	Rain Rate	Solar Rad.	Solar Energy	Hi Solar Rad.	UV Index	UV Dose	Hi UV	Heat D-D	Cool D-D	In Temp	In Hum	In Dew	In Heat	In EMC	In Air Density	ET
11/11/21	4:01p	29.1	29.1	29.1	37	12.9	1.6	SW	0.03	4.8	SSW	29.1	28.4	28.4	---	935.9	0.00	0.0	69	0.10	69	0.9	0.01	0.9	0.000	0.007	26.6	40	11.9	25.9	7.55	1.0720	0.00
11/11/21	4:02p	29.2	29.2	29.1	36	12.6	4.8	SSW	0.08	8.0	SSW	29.2	28.4	28.4	---	935.9	0.00	0.0	69	0.10	69	0.9	0.01	0.9	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0720	0.00
11/11/21	4:03p	29.2	29.2	29.1	36	12.6	8.0	S	0.13	11.3	S	29.2	28.4	28.4	---	935.8	0.00	0.0	67	0.10	67	0.9	0.01	0.9	0.000	0.008	26.7	40	11.9	25.9	7.55	1.0717	0.00
11/11/21	4:04p	29.2	29.2	29.2	36	12.6	3.2	SW	0.05	11.3	SSW	29.2	28.4	28.4	---	935.9	0.00	0.0	69	0.10	69	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0720	0.00
11/11/21	4:05p	29.2	29.2	29.2	37	13.0	1.6	N	0.03	3.2	NNE	29.2	28.5	28.5	---	935.7	0.00	0.0	69	0.10	69	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0718	0.00
11/11/21	4:06p	29.2	29.2	29.2	36	12.6	1.6	N	0.03	3.2	N	29.2	28.4	28.4	---	935.7	0.00	0.0	68	0.10	69	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0718	0.00
11/11/21	4:07p	29.2	29.2	29.2	37	13.0	1.6	N	0.03	3.2	N	29.2	28.5	28.5	---	935.7	0.00	0.0	67	0.10	67	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0718	0.00
11/11/21	4:08p	29.2	29.2	29.2	36	12.6	1.6	NNE	0.03	3.2	NE	29.2	28.4	28.4	---	935.7	0.00	0.0	67	0.10	67	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0718	0.00
11/11/21	4:09p	29.2	29.2	29.2	36	12.6	3.2	SSW	0.05	6.4	SSW	29.2	28.4	28.4	---	935.7	0.00	0.0	67	0.10	67	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0718	0.00
11/11/21	4:10p	29.2	29.2	29.2	36	12.6	3.2	SSW	0.05	6.4	SSW	29.2	28.4	28.4	---	935.7	0.00	0.0	67	0.10	67	0.8	0.01	0.8	0.000	0.008	26.6	40	11.9	25.9	7.55	1.0718	0.00

4.10. PRUEBAS DE CONCEPTO AWS

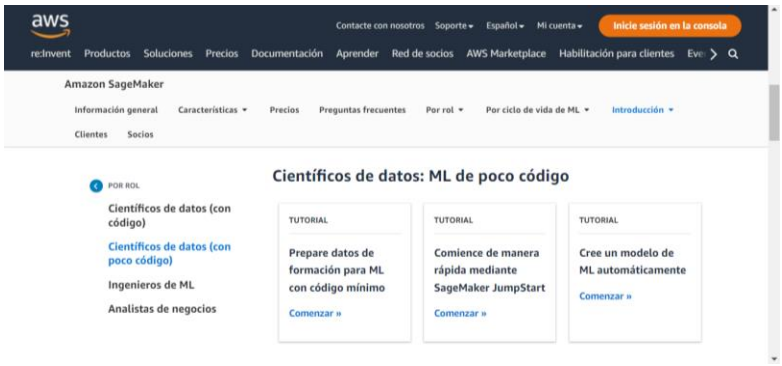
Para utilizar AWS y las herramientas de Machine Learning que nos ofrece, se creo una cuenta y al terminar el registro y los pasos necesarios se observa la siguiente interfaz de inicio:



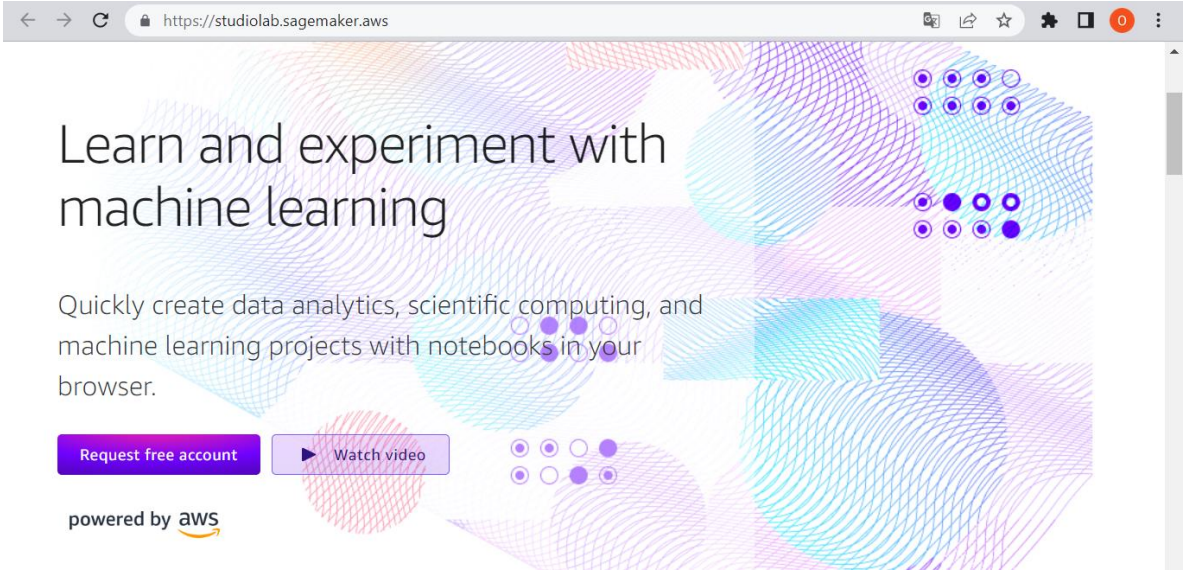
1.-La consola de AWS ofrece un buscador el cual se utiliza para encontrar herramientas que se necesitan:

2.-SageMaker es una herramienta que permite a los desarrolladores crear, entrenar e implementar modelos de aprendizaje automático en la nube, para poder hacer uso de esto se necesitan algunos conocimientos de configuracion y de uso para lo cual AWS ofrece tutoriales de como manejar la herramienta y su respectiva configuracion. Una de las razones por lo que se eligio y principales ventajas de AWS es que esta muy bien documentado en internet.

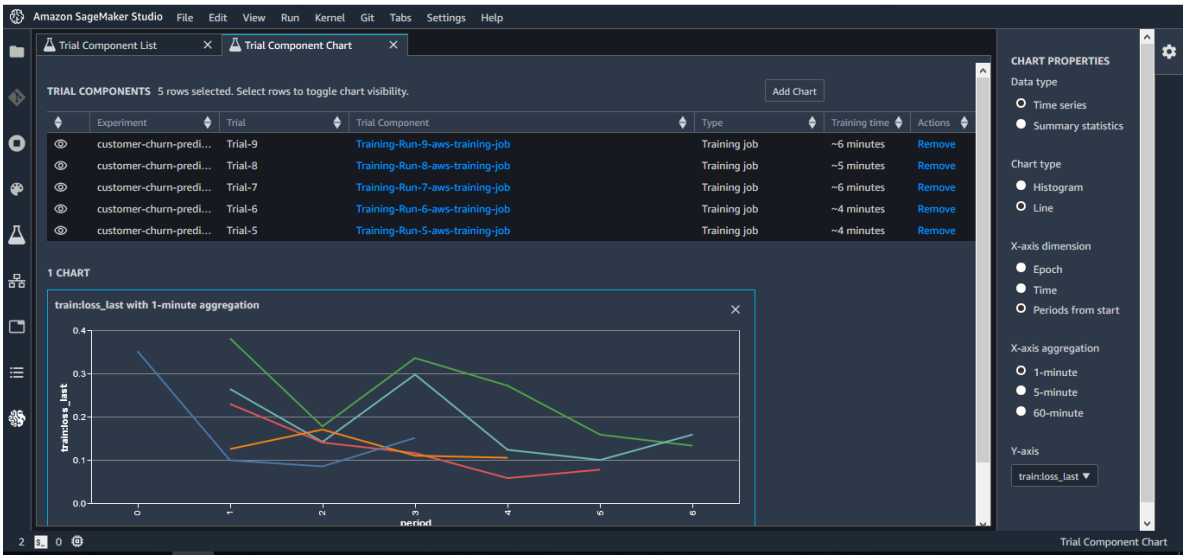
Se realizaron tutoriales en los cuales explica la configuracion y el uso de la herramienta de SageMaker asi como tambien se enriquecio este conocmiento con la informacion encontrada en internet sobre su respectivo uso.



2.-SageMaker Studio Lab es la herramienta que ofrece AWS para la implementación del Machine Learning. De igual manera se realizó la configuración respectiva, así como se investigó más el uso de SageMaker Studio.

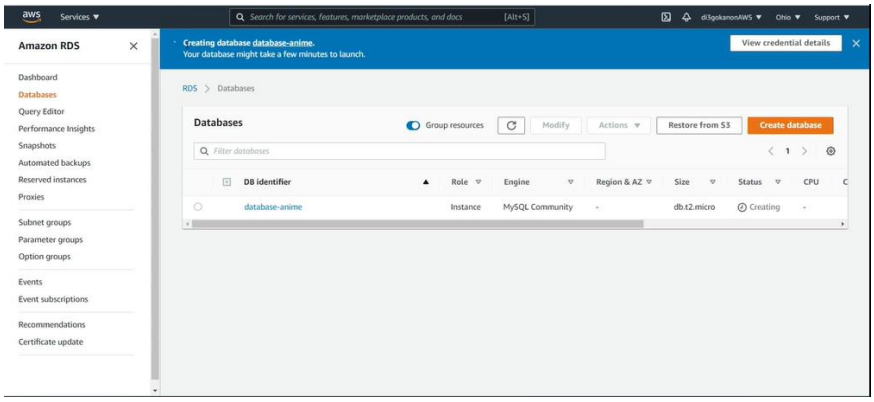


A continuación, un ejemplo de cómo se ve la interfaz principal del SageMaker Studio en el cual se realizó un ejemplo únicamente con el propósito de conocer y aprender el manejo de la herramienta:

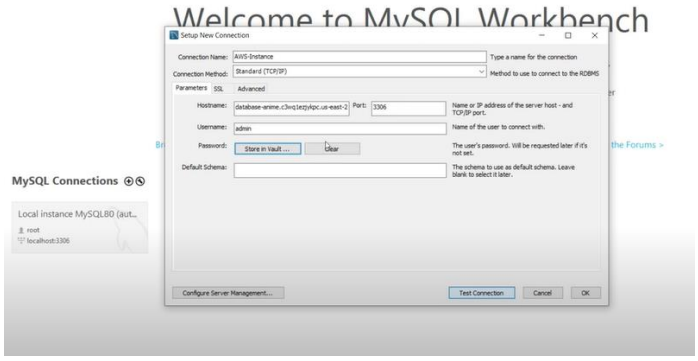


Así mismo para la base de datos, AWS ofrece servicios básicos los cuales con un plan gratis se tuvo la oportunidad de aprender y conocer la configuración respectiva

entre AWS y nuestro MySQL Workbench, a continuación se presenta la pantalla en AWS de la manera en cómo podríamos ver los datos de manera web:



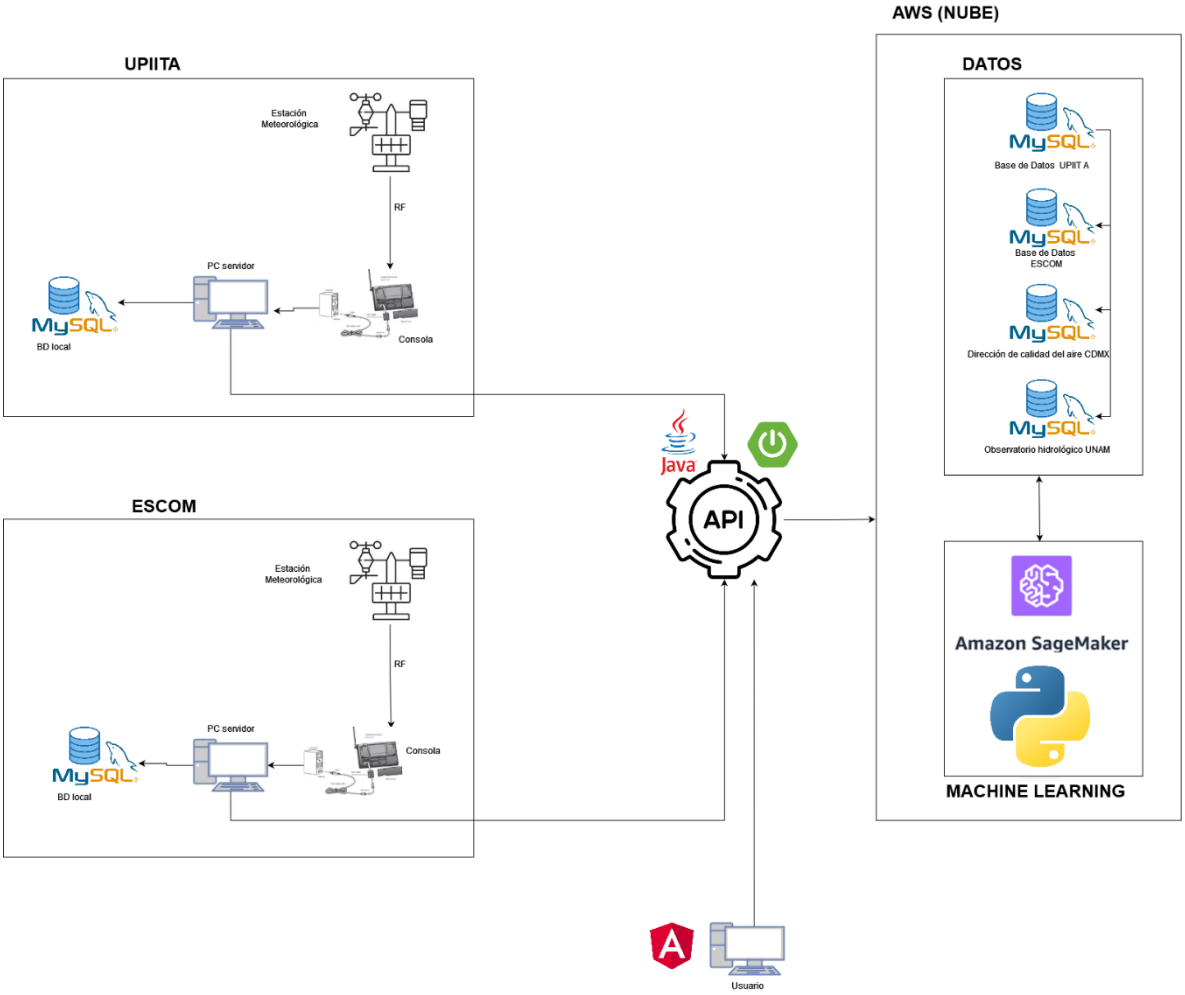
Se realizo la configuración pertinente entre nuestro manejador de base de datos y AWS donde con ayuda de la documentación disponible se configuro con éxito.

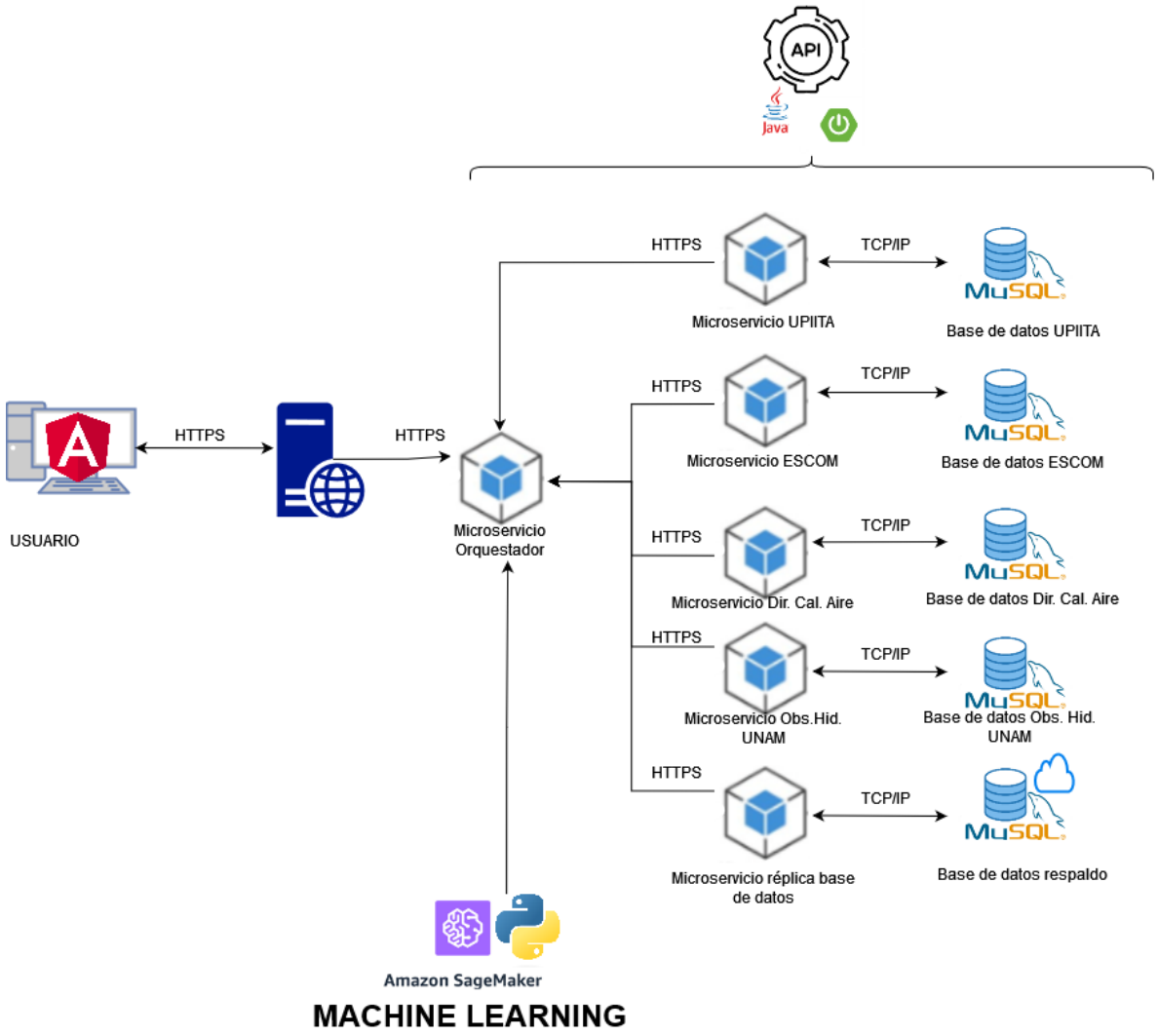


Capítulo 5.
Diseño.

CAPÍTULO 5 DISEÑO

Como anteriormente se mencionó la siguiente imagen muestra la arquitectura del sistema, con el análisis del capítulo anterior se determina lo siguiente para su implementación.

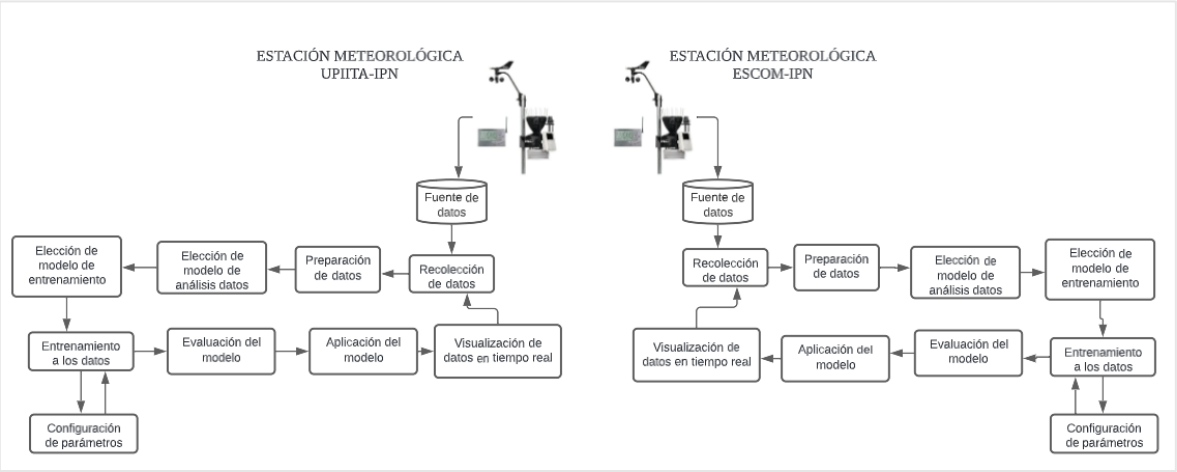




Como muestran las imágenes el desarrollo será implementado con las siguientes herramientas:

- Angular para el desarrollo del FrontEnd.
- SpringBoot (JAVA) para la aplicación de arquitectura de microservicios.
- MySQL para las bases de datos locales y el backup en la nube.
- AWS como nube.
- SageMaker (AWS)

5.1 DISEÑO DEL MODELO MACHINE LEARNING



5.2. DISEÑO DE LA BASE DE DATOS

Como vimos en la sección anterior, para la base de datos, se tiene contemplado los siguientes parámetros para su análisis, la siguiente tabla es la principal debido a que aquí ya se tienen contemplados los parámetros que se intersecan entre las bases de datos; UPIITA, ESCOM, Dirección de Monitoreo Atmosférico-CDMX y por el Observatorio Hidrológico II UNAM.

ID_registro	ID estacion	Fecha	Hora	Temp	Velocidad del viento	Direc. de viento	Lluvia actual	Lluvia diaria	Presión At.	Hum
int	int	Data	Time	int	int	String/varchar	int	int	int	int

5.3. DIAGRAMA DE CASOS DE USO DEL SISTEMA

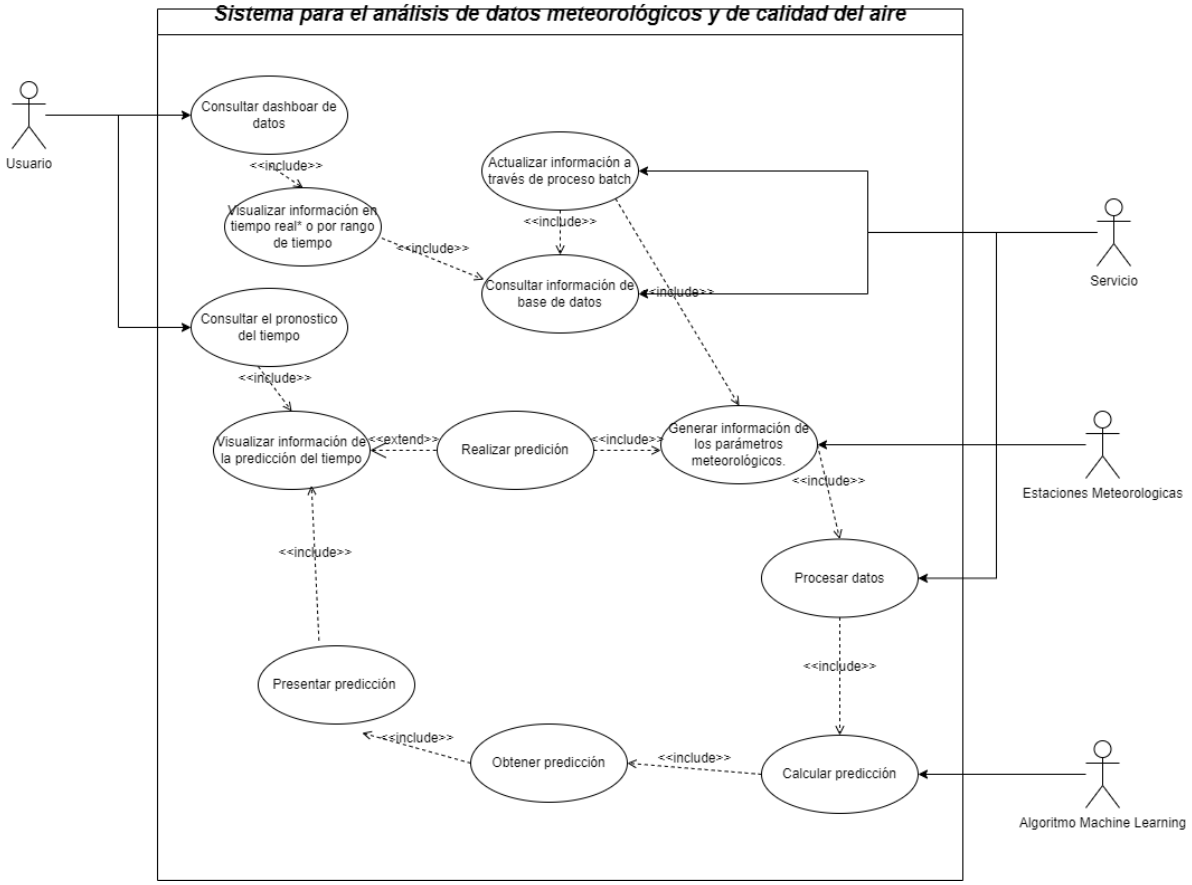


Ilustración 20. Diagrama de casos de uso del sistema general

Nombre	Consultar dashboard de datos
Autor	Aparicio Espinoza Octavio Joel (Usuario) Hernández Castillo Francisco (Usuario) López Tirado Nadia Jaqueline (Usuario)
Fecha	5/01/2023
Descripción: El usuario podrá ingresar a la página web que contendrá la información referente a los datos que se recaban desde la estación meteorológica, en forma de tablas o gráficos para que sea dinámico.	
Precondiciones: Se debe realizar previamente consultas a la base de datos para poder mostrarlos en el dashboard de información.	
Flujo normal: 1) Ingresar al portal web. 2) Visualizar la información	
Flujo alternativo: En caso de que el servicio esté abajó o que la fecha seleccionada esté fuera de lo permitido aparecerá un response con dicho error.	
Postcondiciones: El sistema consultará la información en tiempo real. **Nota:** en el entendido que "tiempo real" se refiere a la actualización cada 8 horas de los datos desplegados en el dashboard.	

Nombre	Consultar dashboard de datos en un rango de tiempo
Autor	Aparicio Espinoza Octavio Joel (Usuario) Hernández Castillo Francisco (Usuario) López Tirado Nadia Jaqueline (Usuario)
Fecha	5/01/2023
Descripción: El usuario podrá ingresar a la página web que contendrá la información referente a los datos que se recaban desde la estación meteorológica, en forma de tablas o gráficos para que sea dinámico.	
Precondiciones: Se debe realizar previamente consultas a la base de datos para poder mostrarlos en el dashboard de información.	
Flujo normal: 3) Ingresar al portal web. 4) Visualizar la información 5) Seleccionar rango de tiempo	
Flujo alternativo: En caso de que el servicio esté abajó o que la fecha seleccionada esté fuera de lo permitido aparecerá un response con dicho error.	
Postcondiciones: El sistema consultará la información que el usuario ingrese.	

Nombre	Consultar análisis de datos (Identificación de patrones)
Autor	Aparicio Espinoza Octavio Joel (Usuario) Hernández Castillo Francisco (Usuario) López Tirado Nadia Jaqueline (Usuario)
Fecha	5/01/2023
Descripción: El usuario podrá decidir si requiere consultar una predicción del estado del tiempo dando clic en dicho apartado.	
Precondiciones: El sistema debe haber cargado los datos obtenidos de los dataset que se utilizarán para la predicción.	
Flujo normal: 1) Ingresar al portal web. 2) Visualizar la información 3) Seleccionar "Consultar pronostico"	
Flujo alternativo: En caso de que no haya sido cargado un dataset con anterioridad devolverá un mensaje de error.	
Postcondiciones: El sistema aplicará el modelo de machine Learning para dicha operación.	

5.4. DIAGRAMAS DE SECUENCIA

5.4.1 CONSULTA DASHBOARD DE DATOS

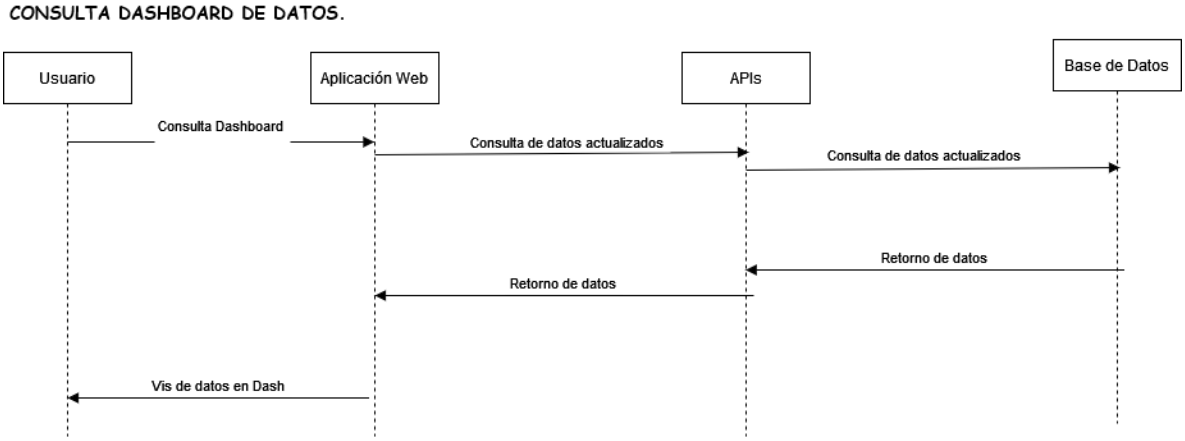


Ilustración 18. Consulta de datos en Dash Board

5.4.2 CONSULTA POR RANGOS DE TIEMPO

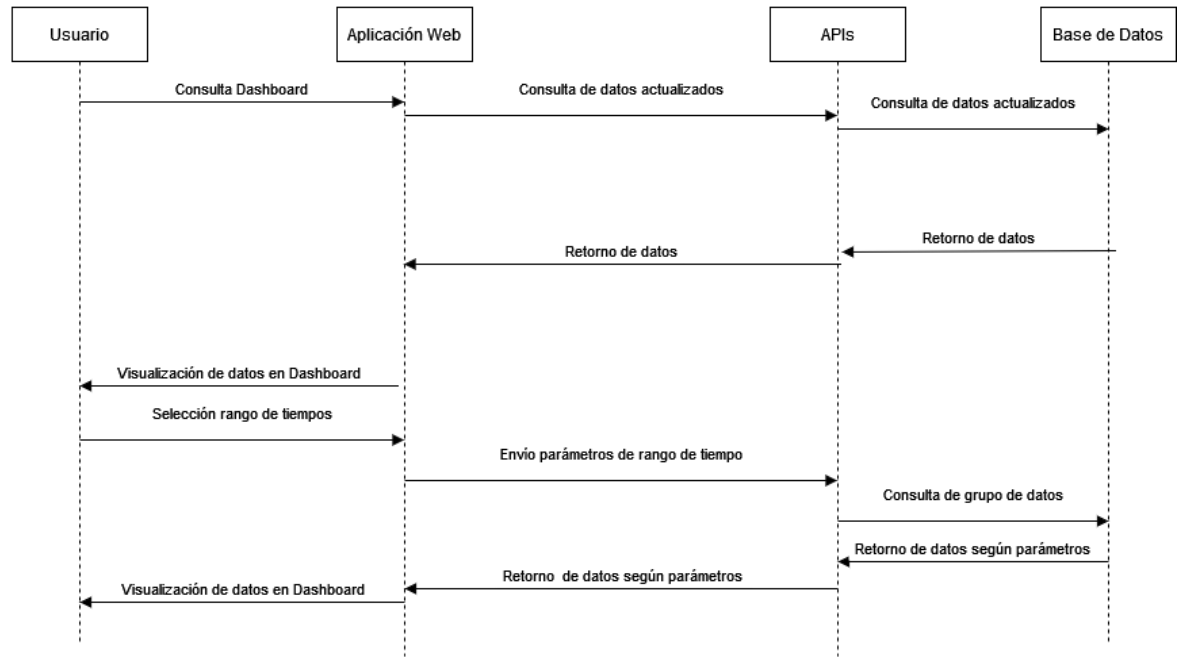


Ilustración 19. Consultas por rangos de tiempo

5.4.3 CONSULTA PRONÓSTICO DE TIEMPO

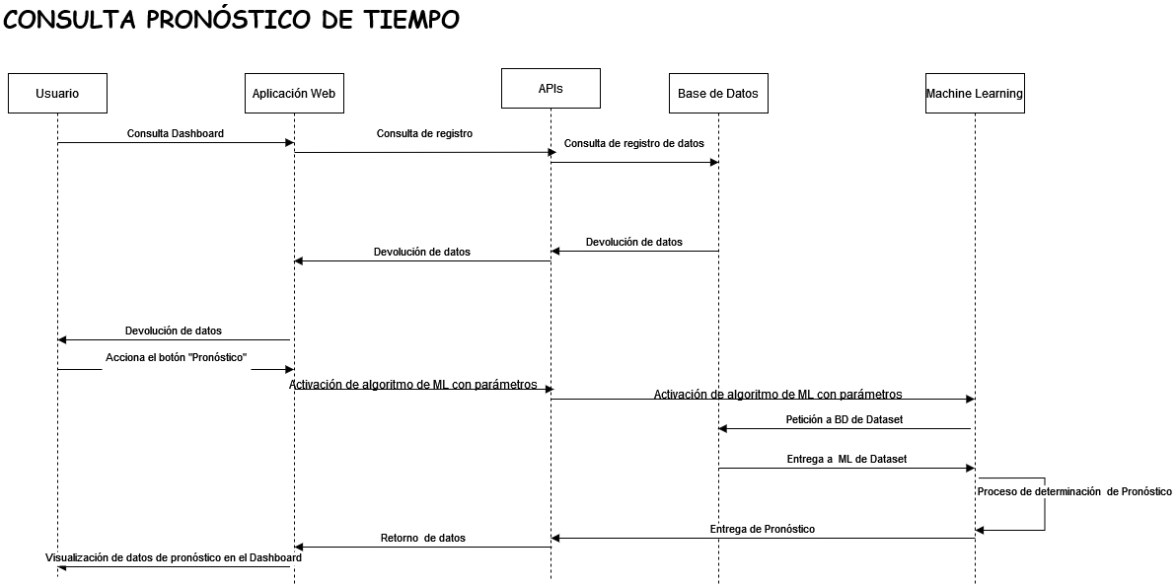


Ilustración 20. Consulta de pronóstico de tiempo

5.5 ESCENARIO DE PRUEBAS

Este proyecto será puesto a prueba bajo la supervisión del Jefe del Laboratorio de Inteligencia Geoespacial y Cómputo Móvil IPN-UPIITA; y será mediante la recolección de datos obtenidos de las dos estaciones meteorológicas que estarán ubicadas una en UPIITA-IPN y la segunda ubicada en instalaciones de ESCOM-IPN después ordenarlos en un modelo de base de datos de manera local o en la nube mediante algoritmos de Inteligencia artificial, manteniendo un registro con su correlación con el comportamiento de calidad del aire para así conocer el microclima de un área específica en este caso la UPIITA-IPN o el área TICOMAN-ZACATENCO con una precisión mayor que usando una App de clima.

En el caso de la estación meteorológica instalada en UPIITA-IPN ha estado instalada desde finales del año 2019 y de forma permanente hasta la actualidad sin embargo por el aislamiento generado por la pandemia no fue posible obtener los datos obtenidos durante todo el año 2021 ni gran parte del año 2020, sin embrago para compensar el vacío de información se tomaran como referencia datos generados por la Dirección de Monitoreo Atmosférico-CDMX y por el Observatorio Hidrológico IIUNAM en los cuales se encuentran acumulados aproximadamente 30 años de información recopilada, de los cuales se tomarán en cuenta al menos 10 años de dicha información.

Para el caso de la estación meteorológica que será instalada en ESCOM-IPN y la cual tiene la misma marca, modelo y características que la que se encuentra en UPIITA-IPN se contempla su instalación para el mes de febrero del presente año 2022 y se espera que dicha instalación sea de forma permanente y forma parte de las actividades de este proyecto terminal.

Las pruebas del Histórico serán llevadas a cabo con la aplicación de Algoritmo de Machine Learning para encontrar patrones en dicha información de comportamiento climatológico y serán comparados para comprobar que la información encontrada en dichos patrones coincida con la realidad y así llevar a cabo predicciones climatológicas confiables.

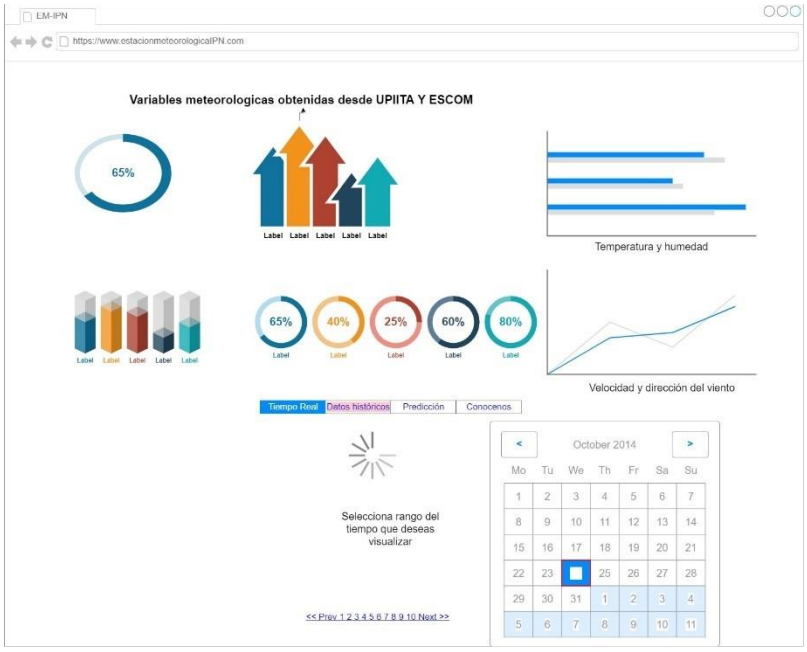
El resultado obtenido se desplegará en un panel (Dashboard), donde se mostrará la información recopilada e interpretada a través del procesamiento del conjunto de datos. De esta manera tendremos en cuenta la eficacia de los componentes de software mencionados en la sección de propuesta de solución.

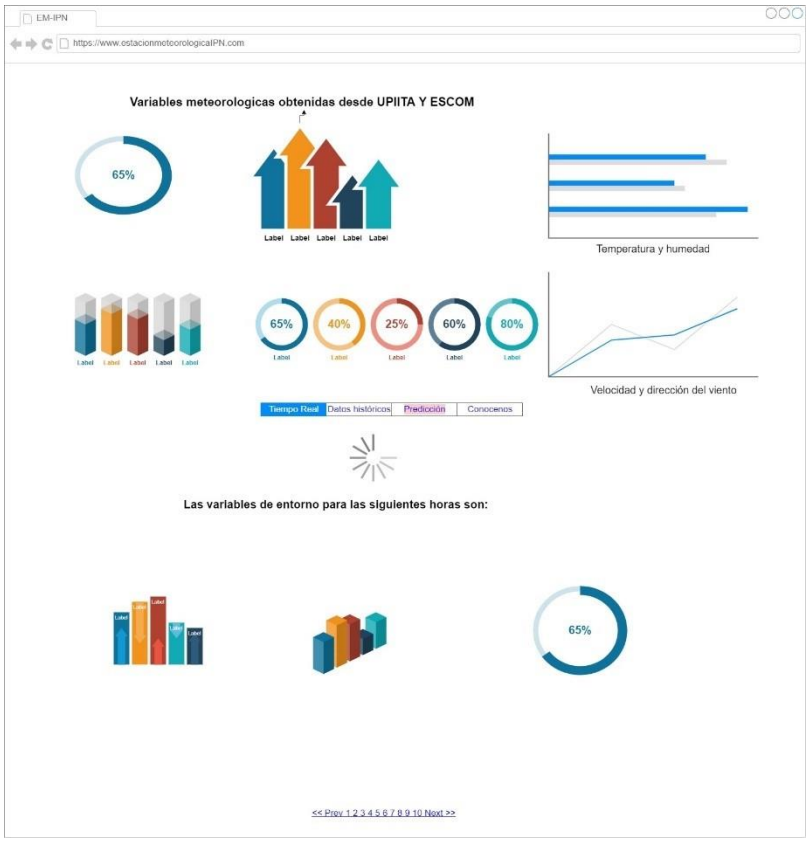
Se tomará en cuenta la funcionalidad del sistema para plantear dichos escenarios, en el entendido que en la lógica de negocio se contempla que se realicen los procesos batch para la actualización de datos que generan las estaciones meteorológicas y con ello se mostrarán los datos a nivel FRONTEND al usuario.

Se podrán probar los escenarios 1 y 2 dado que estos datos se pretenden que sean los datos generados a partir de las estaciones meteorológicas y el escenario 3, se probará a partir del uso de los dataset de apoyo para probar el modelo de Machine Learning.

	Escenario	Descripción
1	Visualizar el dashboard de datos.	1. Ingresar a la página principal. 2. Visualizar los datos generados por las estaciones meteorológicas UPIITA y ESCOM.
2	Visualizar el dashboard de datos dentro de un rango de tiempo.	1. Ingresar a la página principal. 2. Seleccionar “Rango de tiempo”. 3. Visualizar los datos generados por las estaciones meteorológicas UPIITA y ESCOM en el rango de tiempo seleccionado.
3	Identificar patrones (Modelo de Machine Learning)	1. Ingresar a la página principal. 2. Seleccionar “Identificar patrones”. 3. Visualizar los patrones generados por los DataSet de apoyo.

5.6 MUCKUPS





5.7 CRONOGRAMA DE ACTIVIDADES

Tabla 1. Cronograma de actividades para Proyecto Terminal 1.

PROYECTO TERMINAL 2				
No.	**Nombre de la actividad**	**Objetivos**	**Resultados esperados**	**Responsable**
1	Identificación de localización para estación meteorológica	Establecimiento de la mejor área dentro de las instalaciones de ESCOM-IPN para la	Reporte y gráficos sobre la localización óptima de una nueva estación meteorológica	• Alumno 1 • Alumno 2 • Alumno 3

		colocación de una estación meteorológica		
2	Instalación de estación meteorológica en ESCOM-IPN	Montaje de la estructura y elementos que conforman una estación meteorológica dentro de las instalaciones de ESCOM-IPN	Estación meteorológica establecida y funcional.	• Alumno 1 • Alumno 2 • Alumno 3
3	Construcción de arquitectura de microservicios	Construir una arquitectura de microservicios de tal forma que la extracción de datos de la estación meteorológica sea un API a la que se pueda consumir.	Arquitectura de microservicios cuyos datos extraídos sean una API la cual se pueda consumir.	• Alumno 3
4	Desarrollo de la visualización de datos.	Crear una aplicación web para visualizar en gráficos los datos.	Aplicación web donde este un Dashboard que muestre los datos en tiempo real	• Alumno 2
5	Construcción módulo de guardado de datos	Desarrollo dentro de la arquitectura de microservicios de un módulo para el guardado de datos de cada estación meteorológica con su respectiva base de datos	Módulo que guarda datos en su respectiva base de datos.	• Alumno 1

6	Desarrollo del módulo de consulta de comportamiento histórico	Crear una aplicación web para consultar el comportamiento histórico de los datos recopilados el cual contendrá datos históricos.	Aplicación WEB que muestra, tablas, gráficos para que los datos sean de fácil entendimiento.	• Alumno 3
7	Implementación de un análisis de datos para la predicción y detección de valores meteorológicos	Aplicación de algoritmos de Machine Learning para encontrar patrones en la información recabada con el fin de hacer predicciones climatológicas.	Predicciones climatológicas como de temperatura, humedad, probabilidad de lluvia con un alto nivel de fiabilidad.	• Alumno 2
8	Integración y diseño de las bases de datos para almacenar los datos de las estaciones meteorológicas	Diseño de las bases de datos para almacenar los datos de las estaciones meteorológicas, datos abiertos de calidad del aire y de observatorio hidrológico	Bases de datos completas.	• Alumno 1

9	Construcción de una API de microservicios	Desarrollo de una API de microservicios para acceder a los datos de la red de estaciones meteorológicas	API que permite acceder a datos de las estaciones meteorológicas.	• Alumno 3
10	Desarrollo del módulo de extracción de datos	Creación de módulo de extracción de os datos desde la estación de monitoreo con una arquitectura de microservicios para consumirla como API	API con arquitectura de microservicios.	• Alumno 2

11	Desarrollo una aplicación web	Construcción de una aplicación web donde se observe la interfaz de visualización de datos en tiempo real	Aplicación web que muestre los datos históricos de forma gráfica.	• Alumno 1
12	Desarrollo del módulo de exploración de datos	Construcción de un módulo donde se pueda hacer la exploración de datos obtenidos desde las estaciones meteorológicas	Módulo que permita comprender que hay en un conjunto de datos y las características de estos.	• Alumno 3

13	Desarrollo del módulo para la gestión del catálogo	Construcción de un módulo donde se recogerá un catálogo con los fenómenos de alertamiento y monitoreo	Módulo que permite entender fenómenos considerados resaltables del conjunto de datos.	• Alumno 2
14	Diseño el modelo de datos	Documentar el diseño del sistema de datos como un diagrama que represente como fluyen los datos	Gráficos que muestren como fluyen los datos.	• Alumno 1
15	Desarrollo del módulo de alertamiento	Gestionar un catálogo de fenómenos en caso de que se hallen valores atípicos en tiempo real	Gestión de datos antropogénicos para monitoreo en tiempo real.	• Alumno 2
16	Desarrollo del Módulo de exploración y analítica de datos	Explorar datos para indicar valores atípicos dentro de los valores que ya han sido recopilados	Indicar correlaciones entre valores de meteorología y calidad del aire, así como el monitoreo de eventos antropogénicos	• Alumno 1

5.8 CRONOGRAMA DE ACTIVIDADES DIAGRAMA DE GANTT

TAREA 1: Identificación de localización para estación meteorológica

TAREA 2: Instalación de estación meteorológica en ESCOM-IPN

TAREA 3: Construcción de la arquitectura de microservicios

TAREA 4: Desarrollo de la visualización de datos

TAREA 5: Construcción del módulo de guardado de datos

TAREA 6: Desarrollo del módulo de consulta de comportamiento histórico

TAREA 7: Implementación de un análisis de datos para la predicción y detección de valores meteorológicos

TAREA 8: Integración y diseño de base de datos para las estaciones meteorológicas

TAREA 9: Construcción de un API de microservicios

TAREA 10: Desarrollo del módulo de extracción de datos

TAREA 11: Desarrollo de una aplicación web

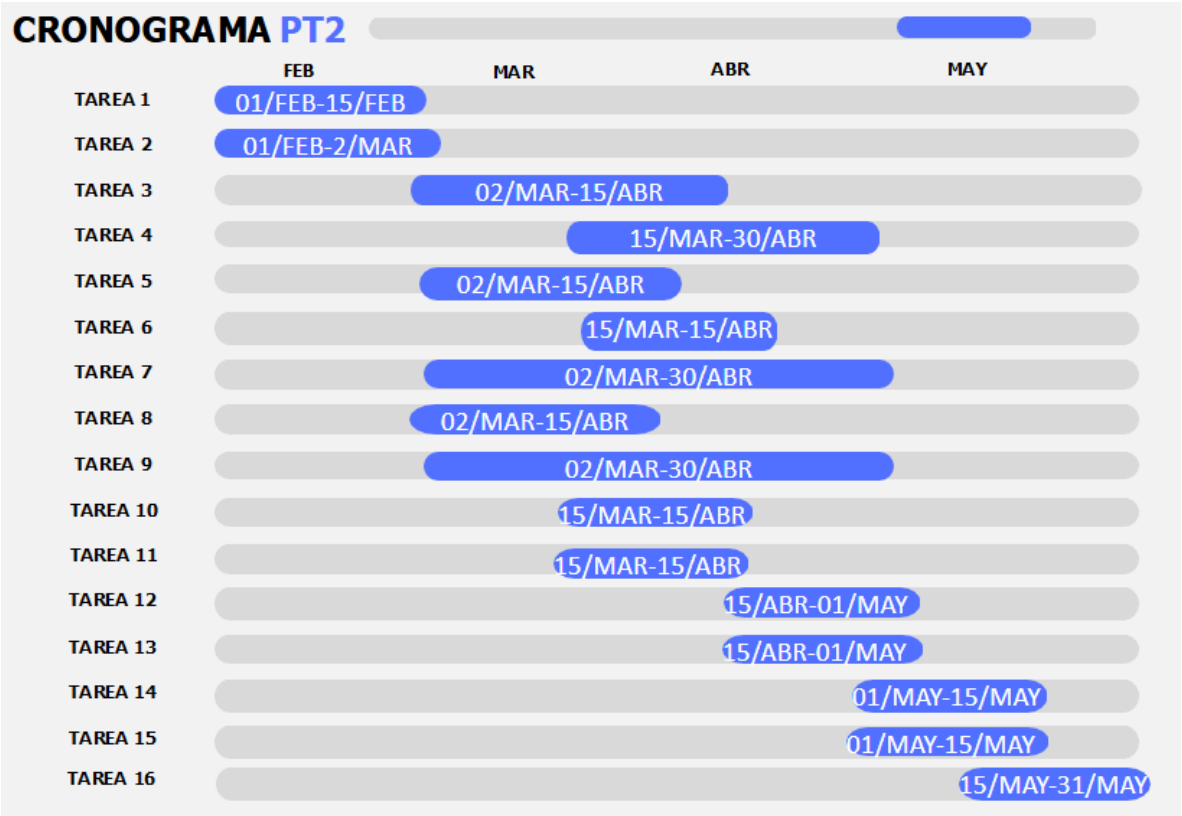
TAREA 12: Desarrollo del módulo de extracción de datos

TAREA 13: Desarrollo del módulo para la gestión del catalogo

TAREA 14: Diseño del modelo de datos

TAREA 15: Desarrollo del módulo de alertamiento

TAREA 16: Desarrollo del módulo de exploración y analítica de datos


CRONOGRAMA PT2
FEB
MAR
ABR
MAY
TAREA 1
01/FEB-15/FEB
TAREA 2
01/FEB-2/MAR
TAREA 3
02/MAR-15/ABR
TAREA 4
15/MAR-30/ABR
TAREA 5
02/MAR-15/ABR
TAREA 6
15/MAR-15/ABR
TAREA 7
02/MAR-30/ABR
TAREA 8
02/MAR-15/ABR
TAREA 9
02/MAR-30/ABR
TAREA 10
15/MAR-15/ABR
TAREA 11
15/MAR-15/ABR
TAREA 12
15/ABR-01/MAY
TAREA 13
15/ABR-01/MAY
TAREA 14
01/MAY-15/MAY
TAREA 15
01/MAY-15/MAY
TAREA 16
15/MAY-31/MAY

REFERENCIAS

[1] Montalvo Lezema, B. (2013) Prototipo Didáctico de una Estación Meteorológica Monitoreada a Distancia [Tesis Licenciatura, Instituto Politécnico Nacional]. https://tesis.ipn.mx/jspui/bitstream/123456789/13084/1/Berenice%20Montalvo%20 Lezama.pdf

[2] Eviatar Zerubavel. (2001) The Clockwork Muse: A Practical Guide to Writing Theses, Dissertations, and Books. Harvard University Press. https://books.google.com.mx/books?id=dePkAwAAQBAJ&pg=PT6&lpg=PT6&dq=The+Clockwork+Muse+editorial&source=bl&ots=xe1uqduqut&sig=ACfU3U2MBHod 584IlAI_PsOzaHmI3sfbmA&hl=es 419&sa=X&ved=2ahUKEwiU86rq8fn2AhWEDkQIHcuBCu0Q6AF6BAgREAM#v=onepage&q=The%20Clockwork%20Muse%20editorial&f=false

[3] Davis Instruments. Wireless Vantage Pro2™ Console Receiver. (06 de septiembre de 2021). https://www.davisinstruments.com/collections/vantage-pro2-console/products/vantage-pro2-consolereceiver

[4] Davis Instruments. WeatherLink® USB Data Logger.(06 de marzo de 2022). https://www.davisinstruments.com/products/weatherlink-windows usb?_pos=3&_sid=414740da1&_ss=r

[5] WeeWX. Open-source software for your weather station. https://weewx.com/

[6] "Classification Algorithm in Machine Learning - Javatpoint". www.javatpoint.com.https://www.javatpoint.com/classification-algorithm-in machine

learning

[7] S. S. Shwartz y S. B. David, “Understanding Machine Learning from Theory to Algorithms,”. R.U:Cambridge University press, 2014, https://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/understanding machine-learning theory-algorithms.pdf

[8] Universidad Nacional Autónoma de México, Observatorio Hidrológico en Tiempo Real (18 de junio de 2018). https://www.oh-iiunam.mx/

[9] Universidad Nacional Autónoma de México, Red Universitaria de Observatorios Atmosféricos. https://www.ruoa.unam.mx/

[10] Oracle. Learn Architect Microservice. https://docs.oracle.com/es/solutions/learn-architect-microservice/index.html#GUID BDCEFE30-C883-45D5-B2E6-325C241388A5

[11] SensorGo MKT. Estaciones Meteorológicas. ¿Qué son y cómo funcionan? (15 de septiembre de 2020). https://sensorgo.mx/estaciones-meteologicas/

[12] Salgado Álvarez, N. (2018) Diseño de una herramienta hidro-informática para el análisis de calidad de datos de estaciones meteorológicas automatizadas. [Tesis Licenciatura, Universidad autónoma del estado de México]. http://ri.uaemex.mx/bitstream/handle/20.500.11799/95372/TESIS.pdf?sequence=1 &isAllowed=y

[13] Davis Instruments. (SPEC SHEET) Vantage Pro2 Wireless Stations Specifications.(04 de mayo de 2022). https://support.davisinstruments.com/article/1fdqvb7bj0-spec-sheet-vantage-pro-2- wireless-stations-specifications

[14] Adobe. ¿Qué son las aplicaciones web y las páginas web dinámicas? Usos comunes de las aplicaciones Web (03 de mayo de 2021). https://helpx.adobe.com/es/dreamweaver/using/web-applications.html

[15] BBVA. Inteligencia Artificial. Machine learning: ¿qué es y cómo funciona? (08 de noviembre de 2019). https://www.bbva.com/es/machine-learning-que-es-y como-funciona/

[16] Dirección de Monitoreo atmosférico. Pronóstico de calidad del aire y meteorológico para la CDMX (27 de mayo de 2022). http://www.aire.cdmx.gob.mx/pronostico-aire/pronostico-meteorologico.php

[17]Observatorio Hidrológico II UNAM. Acuífero (27 de mayo de 2022). https://www.oh-iiunam.mx/acuifero.html

[18] IBM Cloud Education. ¿Qué es una API REST? Principios de diseño de REST (06 de abril de 2021). https://www.ibm.com/mx-es/cloud/learn/rest-apis

[19] IBM Cloud Education. Microservicios. Servicios de microservicios y nube (30 de marzo de 2021). https://www.ibm.com/mx-es/cloud/learn/microservices

[20] *MySQL*. (s. f.). https://www.mysql.com/

[21] Data Science - Machine Learning for Data Scientists - Amazon Web Services. (s. f.).Amazon Web Services, Inc. https://aws.amazon.com/es/sagemaker/data-scientist/

[22] *AWS | Cloud Computing - Servicios de informática en la nube*. (s. f.). Amazon Web Services, Inc. https://aws.amazon.com/es/

[23] *Java Platform Evolution - Dev.java*. (s. f.). Dev.java: The Destination for Java Developers. https://dev.java/evolution/

[24] *Angular*. (s. f.). https://angular.io/guide/what-is-angular

[25] *Spring Framework*. (s. f.). spring. https://spring.io/projects/spring-framework

[26] MICROSOFT. (18/11/2022). *Componente Regresión lineal*. MOCRISOFT. LEARN Recuperado de https://learn.microsoft.com/es-es/azure/machine-learning/component-reference/linear-regression

[27] Juan Ignacio Bagnato. (29 de Noviembre 2018). *Cómo funcionan las Convolutional Neural Networks*. Madrid. INAIT Recuperado de https://www.aprendemachinelearning.com/como-funcionan-las-convolutional-neural-networks-vision-por-ordenador/

[28] Google developers. (18 de Noviembre de 2022). *¿Qué es el agrupamiento en clústeres?*. Google developer network. machine learning Recuperado de https://developers.google.com/machine-learning/clustering/overview

[29] Mathworks. (15 de mayo de 2021). *Support Vector Machine (SVM)*. Mathworks. Mathworks discovery Recuperado de https://la.mathworks.com/discovery/support-vector-machine.html