

Herramienta computacional para el análisis de la vibración en motores eléctricos alimentada mediante datos de una simulación digital

Gerardo Campos y José Cortez

Universidad Nacional Experimental Politécnica "Antonio José de Sucre"

6 de abril de 2022

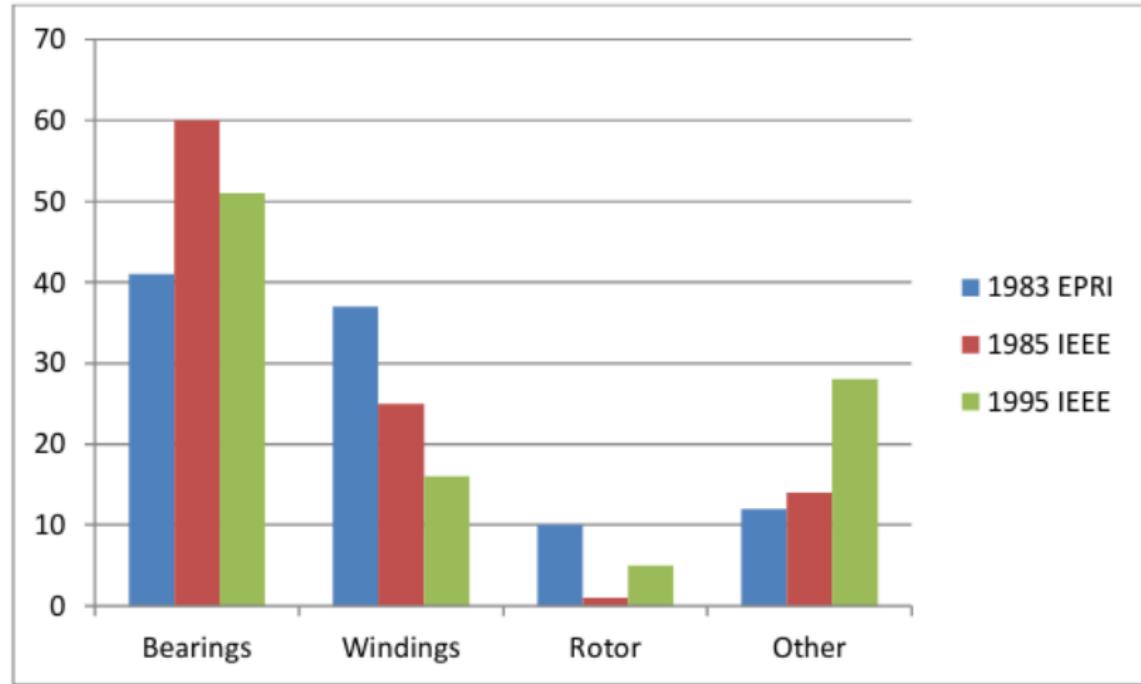


Planteamiento del problema

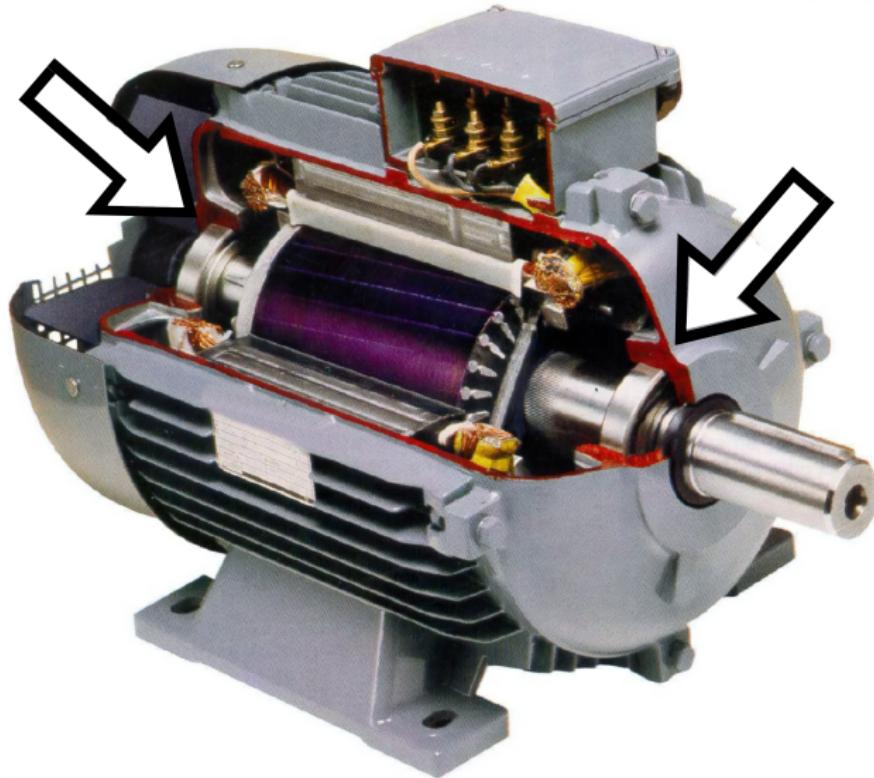
Planteamiento del problema



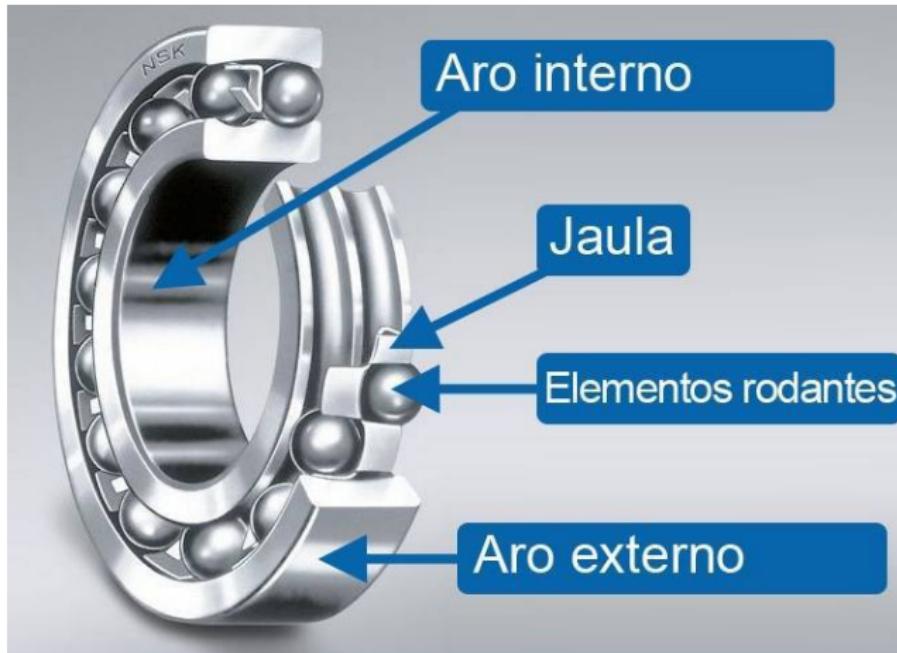
Planteamiento del problema



Planteamiento del problema

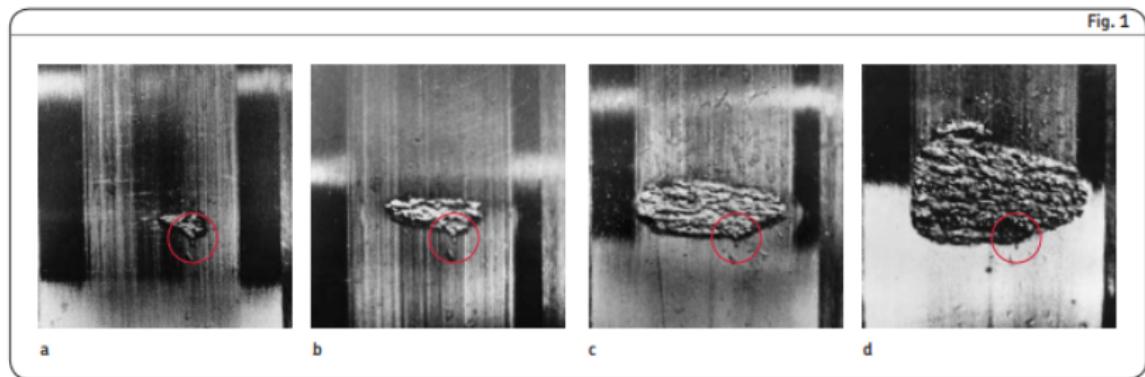


Planteamiento del problema



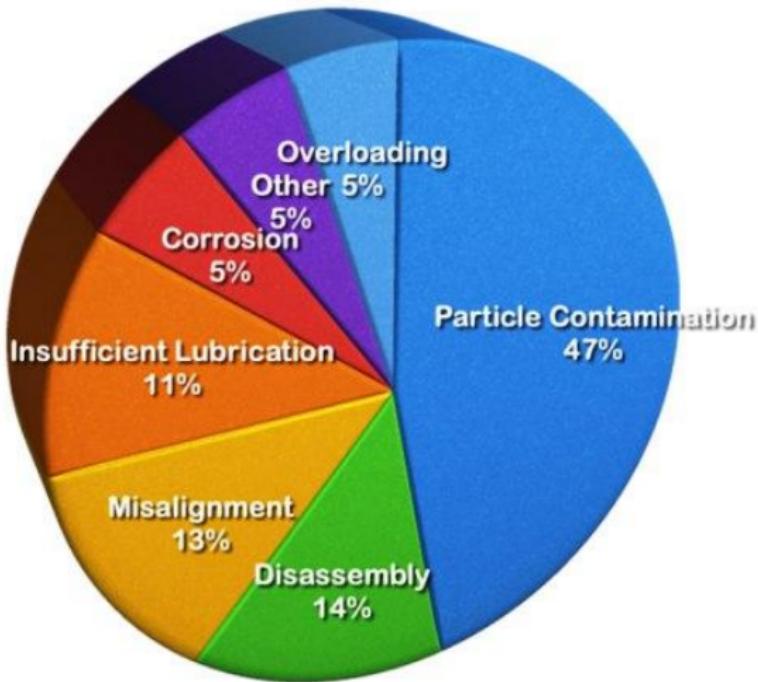
Planteamiento del problema

Fig. 1



Planteamiento del problema

Causes of Bearing Failures



Planteamiento del problema



Objetivos

Objetivo General

Desarrollar una herramienta computacional para el análisis de la vibración en motores eléctricos, mediante la simulación digital de un acelerómetro, con la finalidad de que un operador determine averías y sus causas.

Objetivos específicos

Justificar la escogencia de las herramientas y lenguajes a utilizar en las diferentes etapas que requiere la simulación.

Objetivos específicos

Generar un modelo estadístico de la vibración en motores eléctricos con distinto nivel de daño utilizando una base de datos de la salida de un acelerómetro digital.

Objetivos específicos

Generar un modelo estadístico de la vibración en motores eléctricos con distinto nivel de daño utilizando una base de datos de la salida de un acelerómetro digital.

Elaborar una base de datos con información obtenida del modelo estadístico para alimentar los niveles de análisis de la herramienta.

Objetivos específicos

Realizar análisis de fallas en frecuencia, a partir de la salida del modelo del acelerómetro.

Objetivos específicos

Mostrar la información solicitada de acuerdo al nivel de análisis seleccionado. según sea: Vista Principal, Vista Específica o Vista Exhaustiva.

Objetivos específicos

Mostrar la información solicitada de acuerdo al nivel de análisis seleccionado. según sea: Vista Principal, Vista Específica o Vista Exhaustiva.

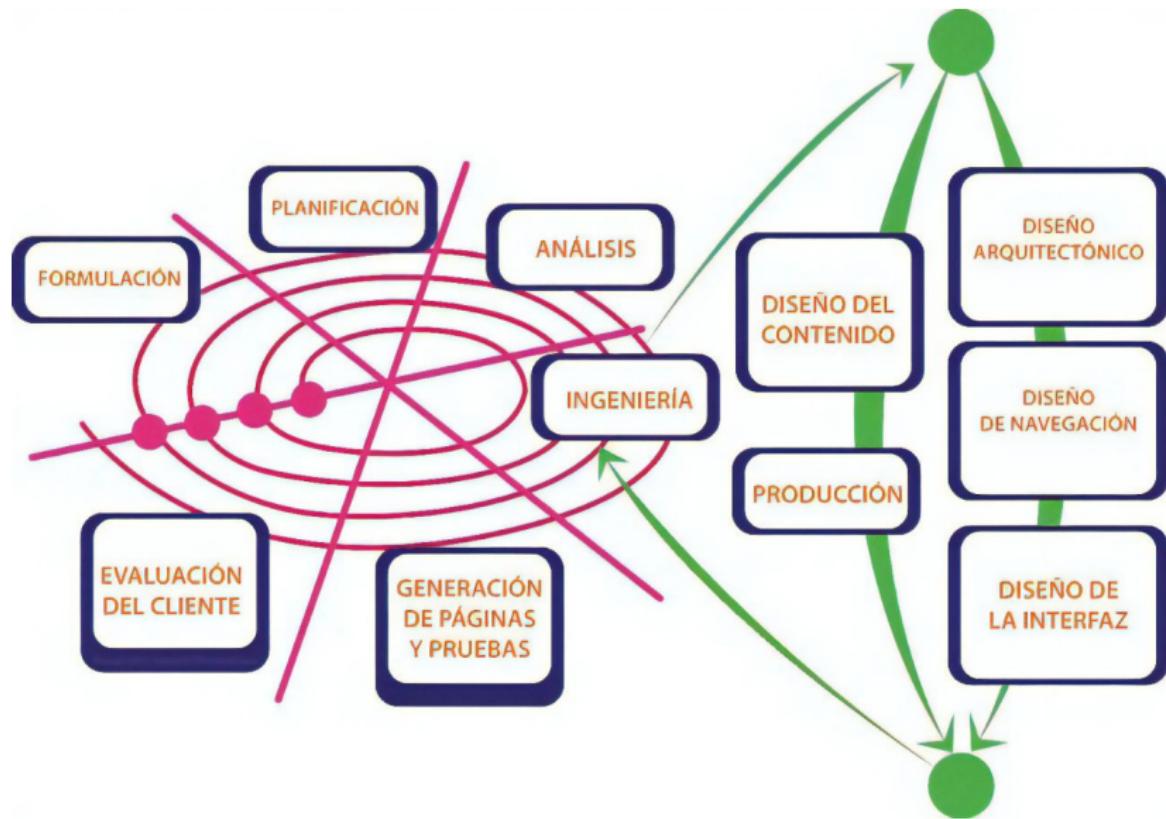
Elaboración de una página Web para facilitar la utilización del sistema

Objetivos específicos

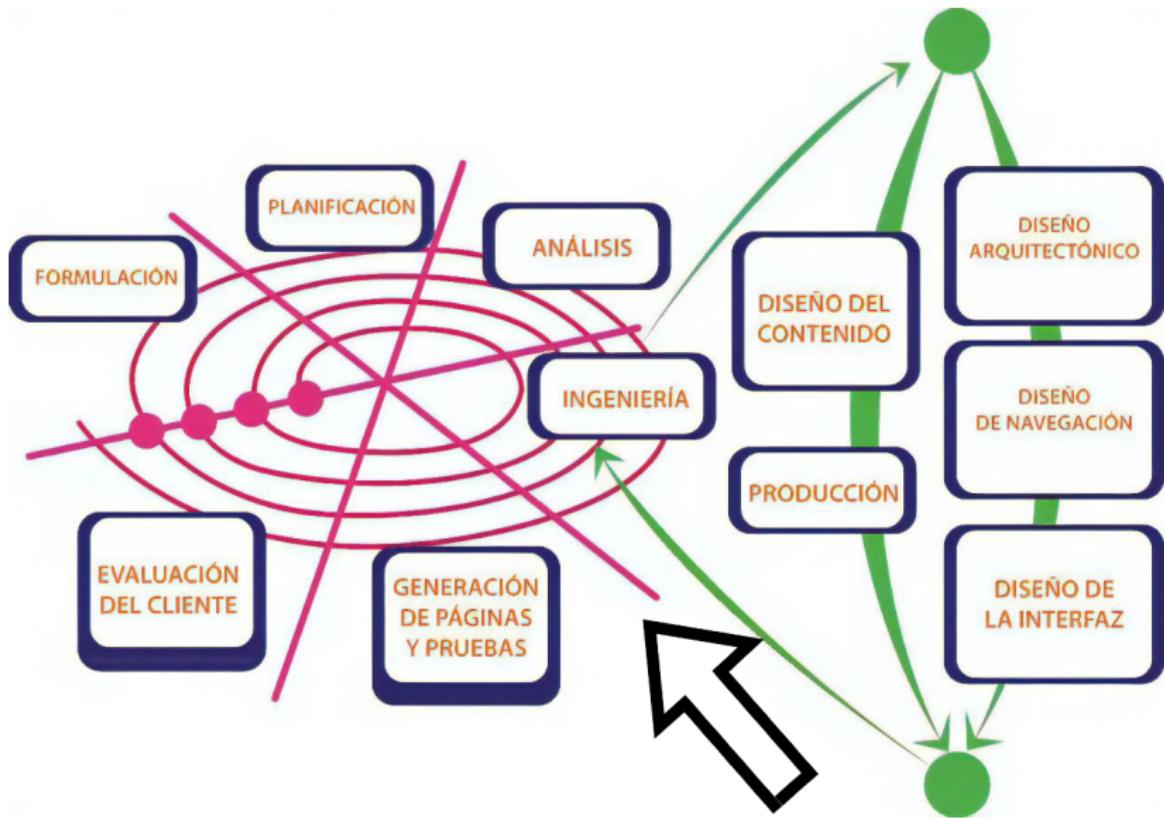
Comprobar los resultados de la herramienta de análisis.

Metodología

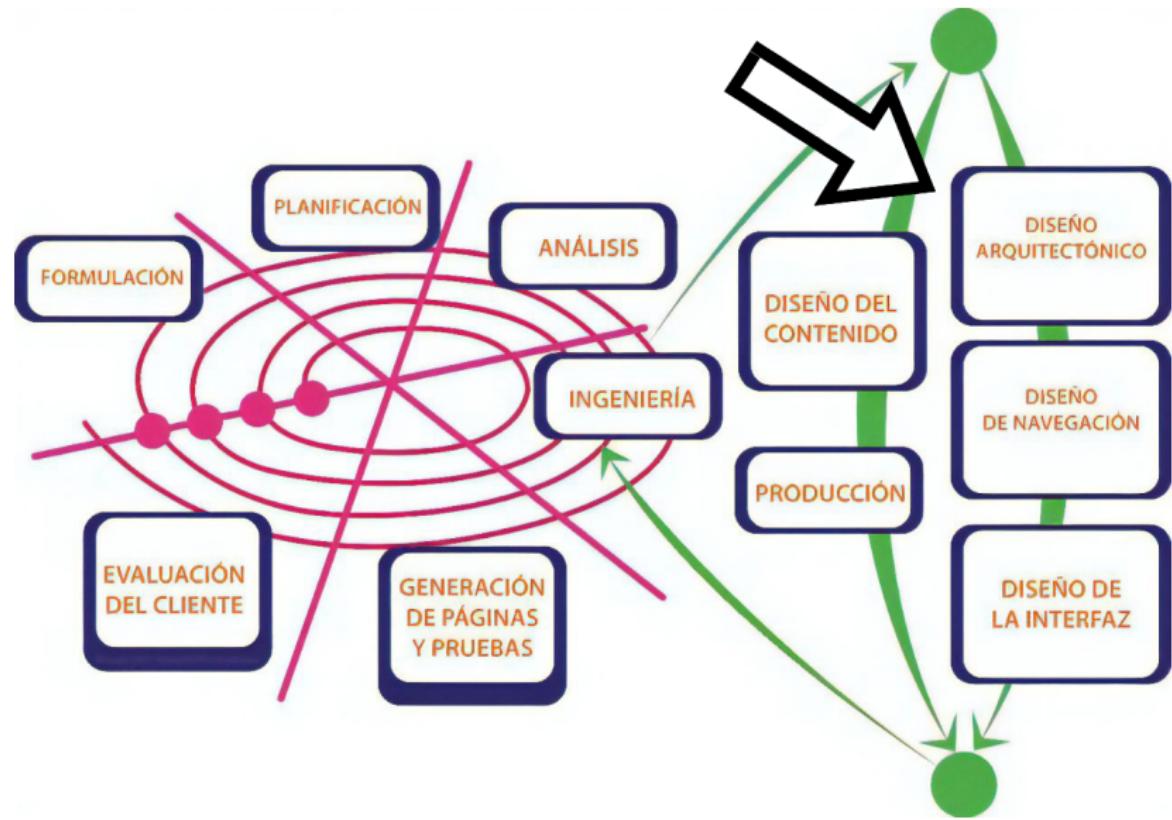
Metodología Web



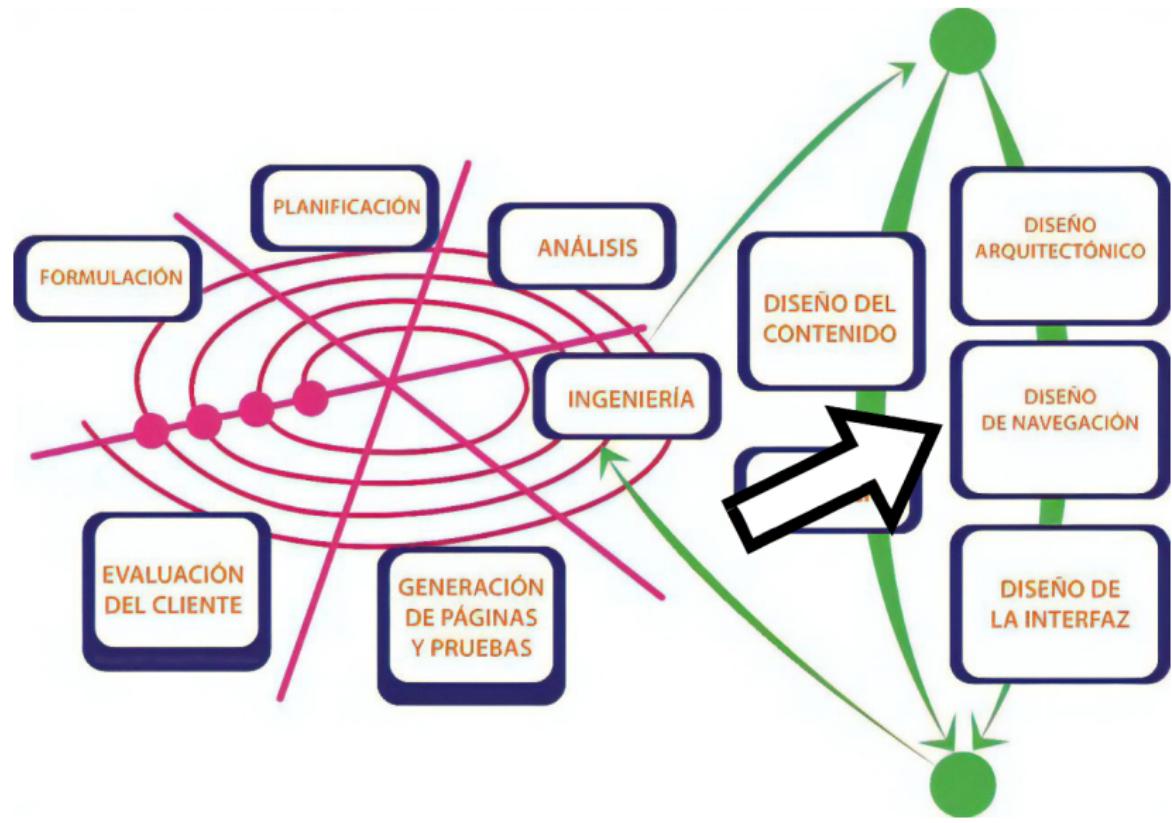
Metodología Web



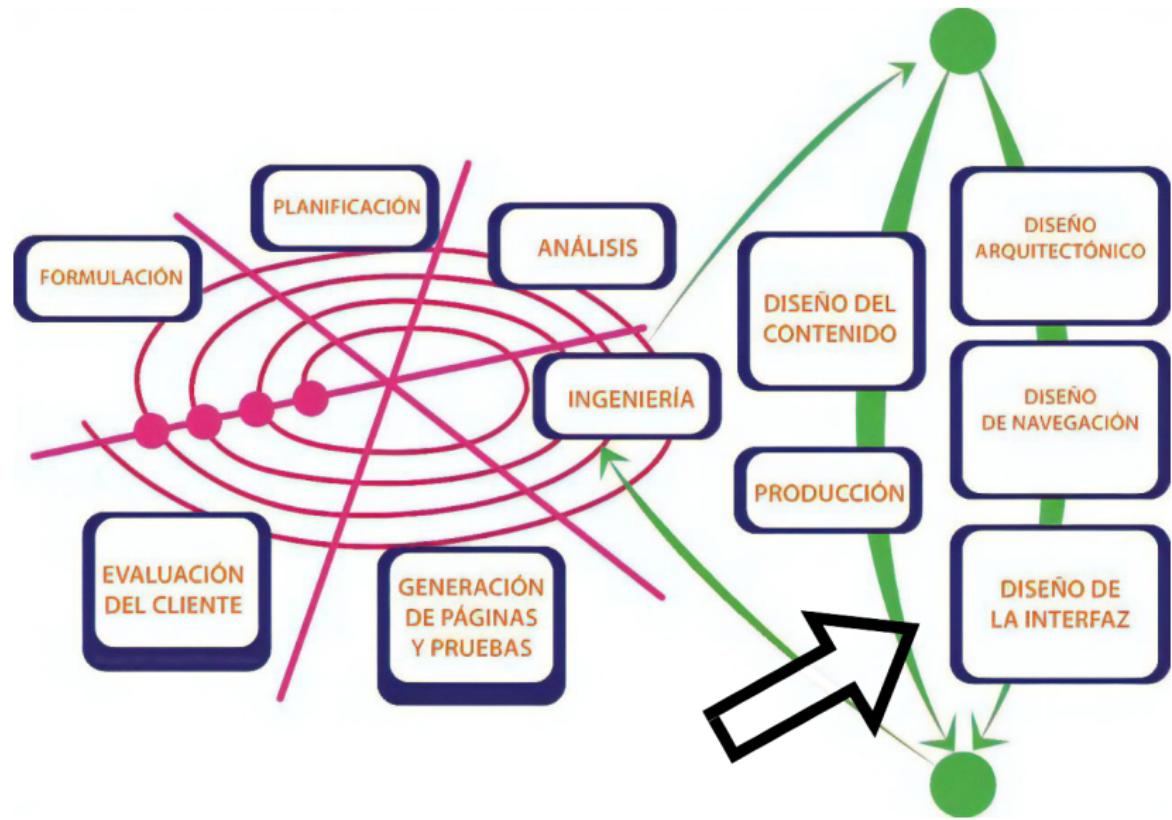
Metodología Web



Metodología Web



Metodología Web



Metodología Web

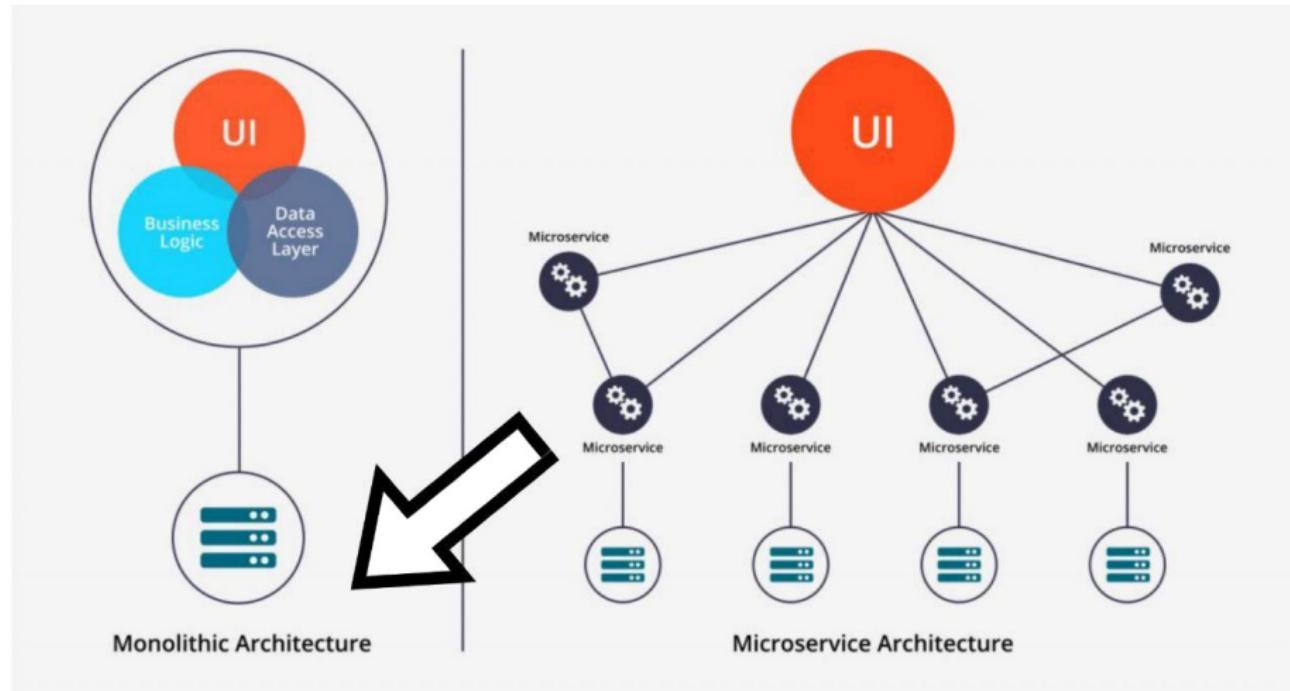
Fase de desarrollo.

Metodología Web

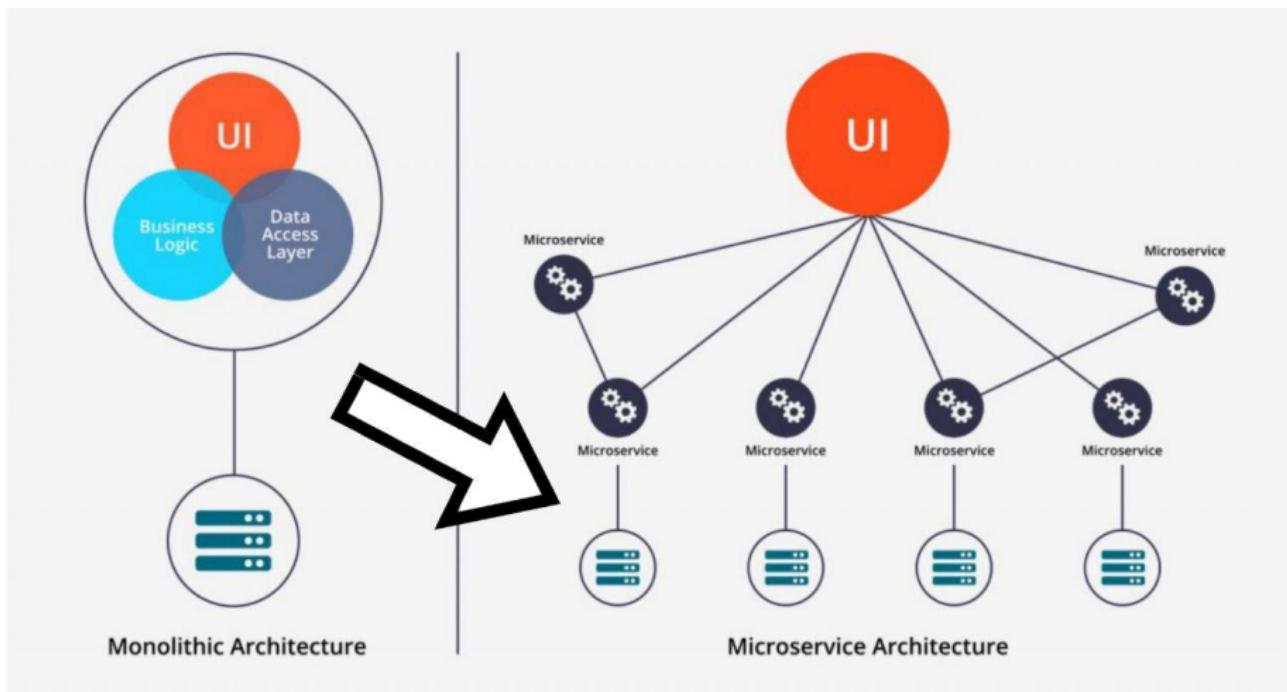
Fase de desarrollo.

Fase de prueba.

Microservicios

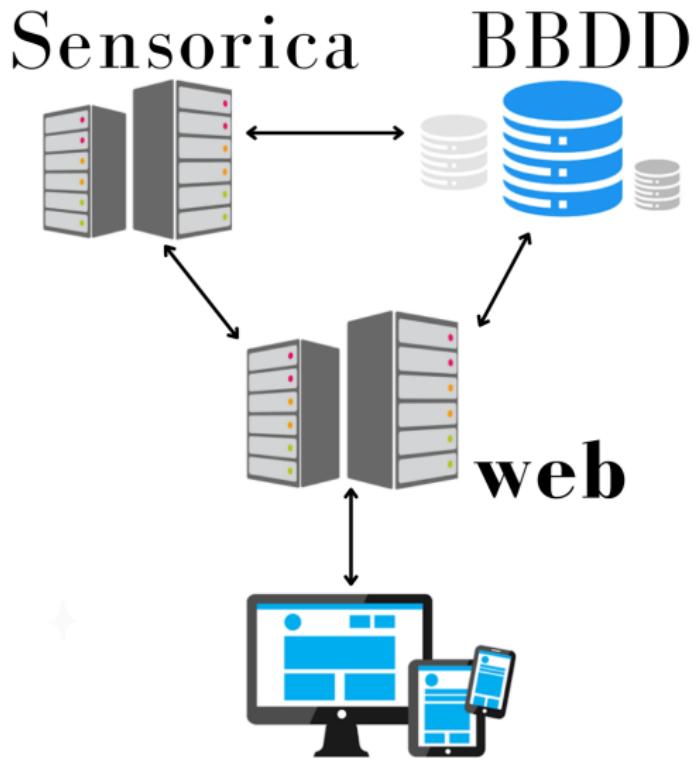


Microservicios



Resultados

Estructura de la aplicación



Lenguajes y Herramientas

The screenshot shows the Atom code editor interface. On the left, the file tree displays project files including `index.html`, `input.css`, `main.css`, `highlight.css`, and various image files like `bg2.png`. The main editor area shows two tabs: `main.css` and `page.html`. The `main.css` tab contains CSS rules for a `#box` element, including background images and padding. The `page.html` tab contains the full HTML structure of a page with meta tags, title, and links to CSS and JS files. On the right, the Atom package manager is open, showing the "Install Packages" section with the "Featured Packages" heading. It lists several packages: `color-picker` by thomaslindstrom (version 2.0.12), `merge-conflicts` by smashwilson (version 1.3.6), `minimap` by atom-minimap (version 4.13.4), and `pigments` by abe33 (version 0.15.0). Each package entry includes a brief description, its version number, the number of downloads, and buttons for settings, uninstall, and disable.

Lenguajes y Herramientas



Análisis Exploratorio







Análisis Exploratorio



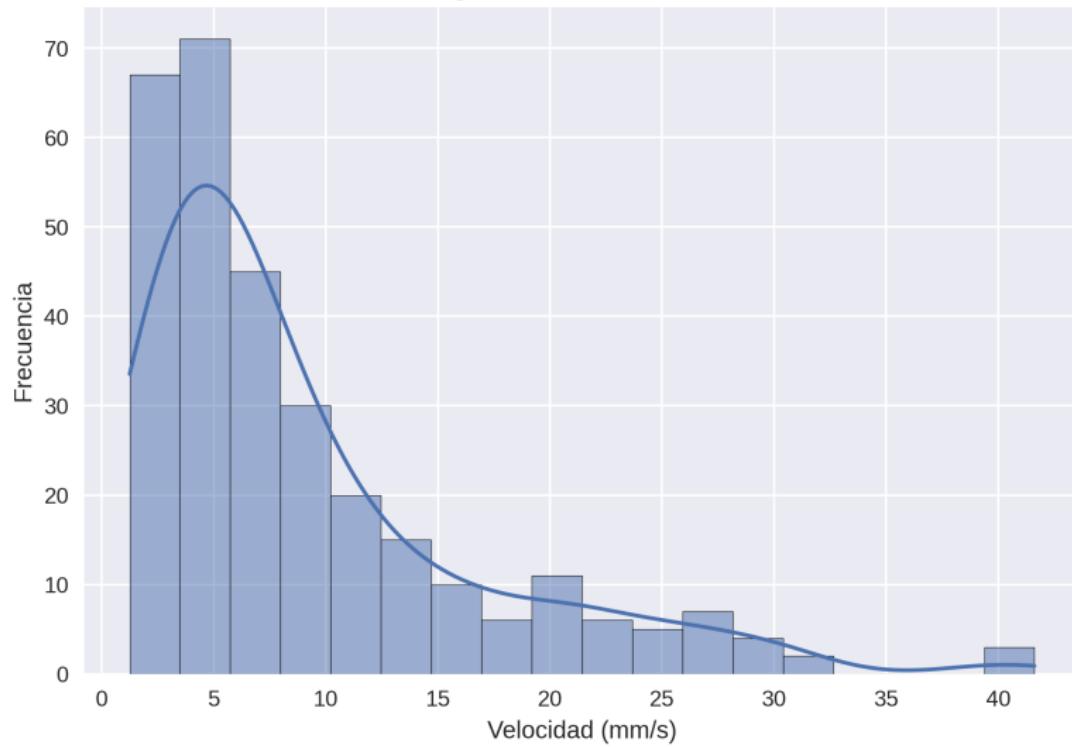
SciPy

matplotlib



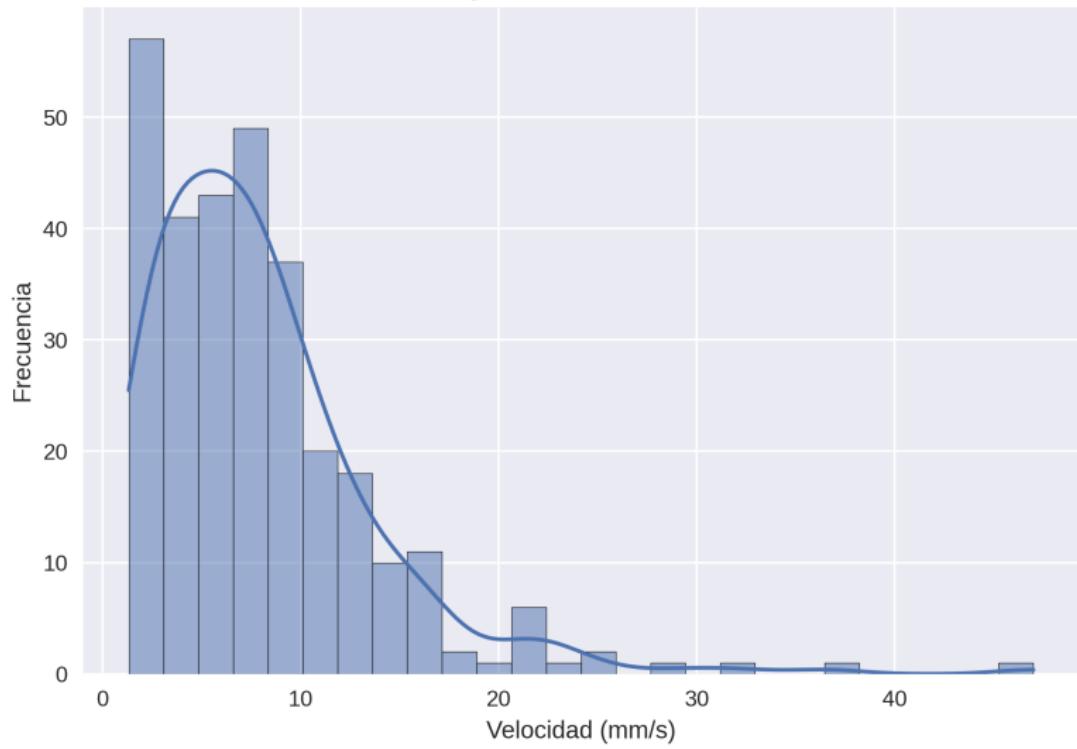
Análisis Exploratorio

Histograma velocidad horizontal



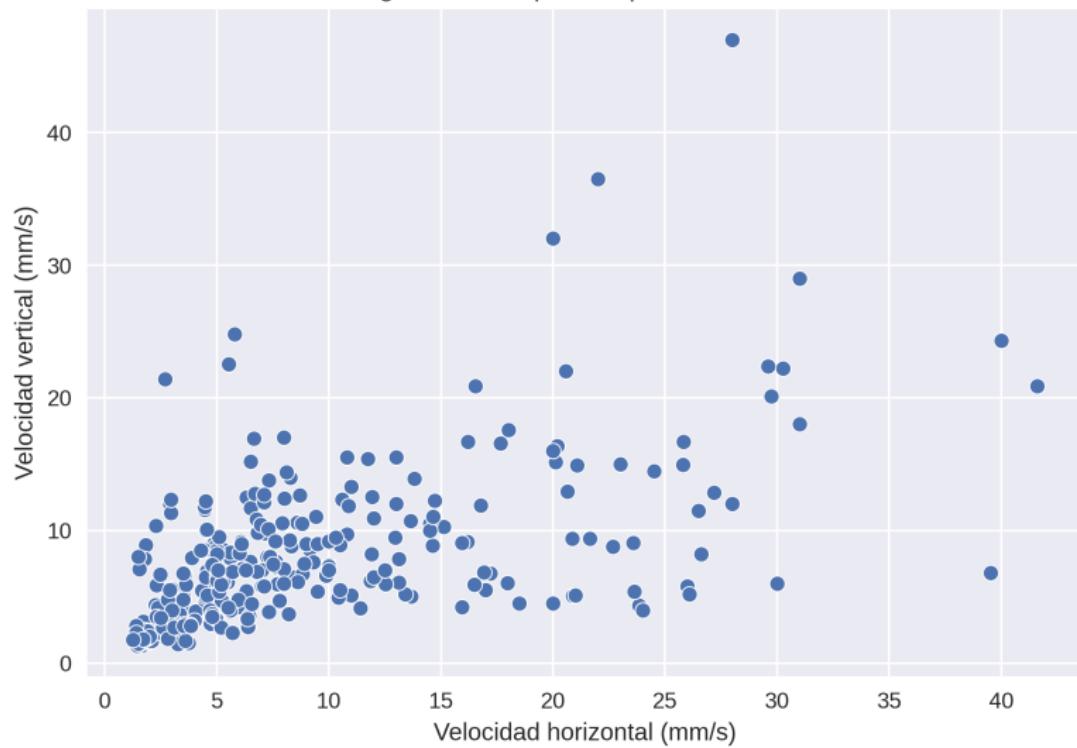
Análisis Exploratorio

Histograma velocidad vertical



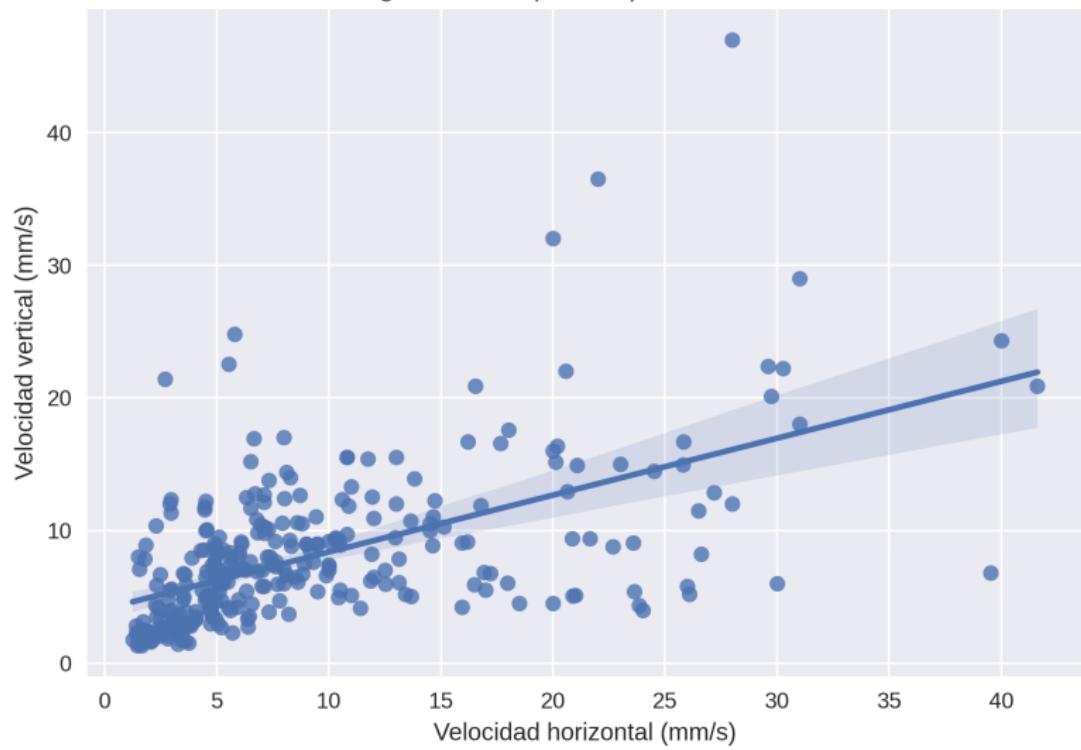
Análisis Exploratorio

Diagrama de dispersion para velocidades

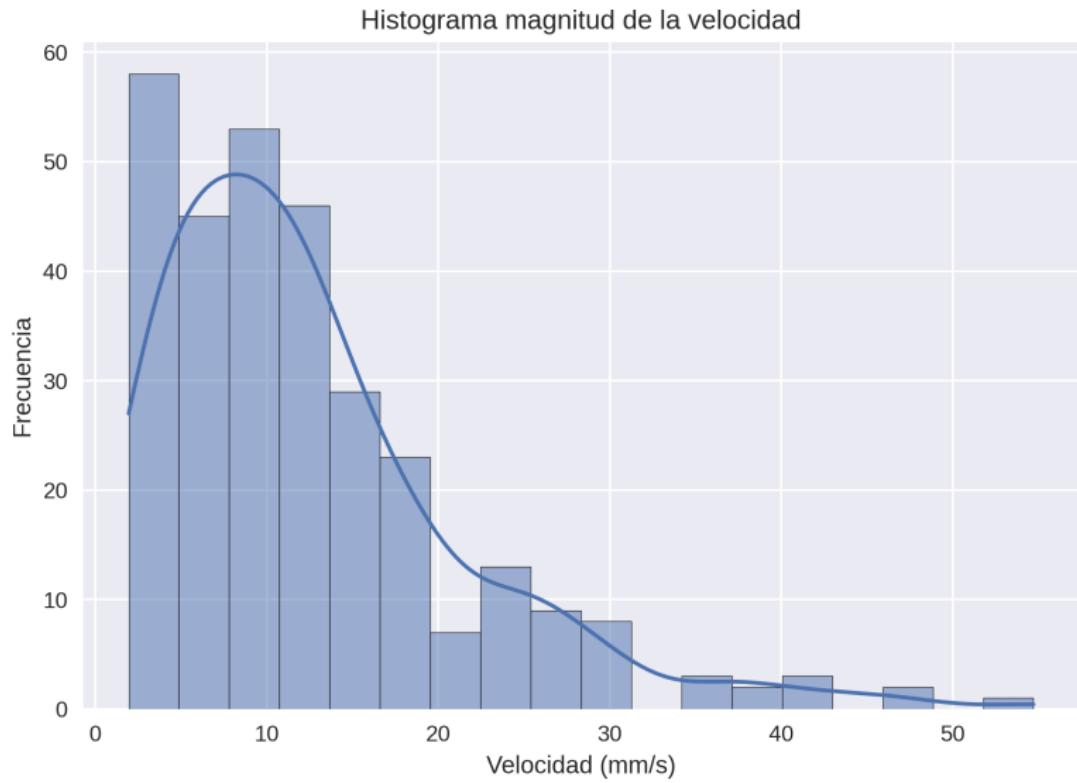


Análisis Exploratorio

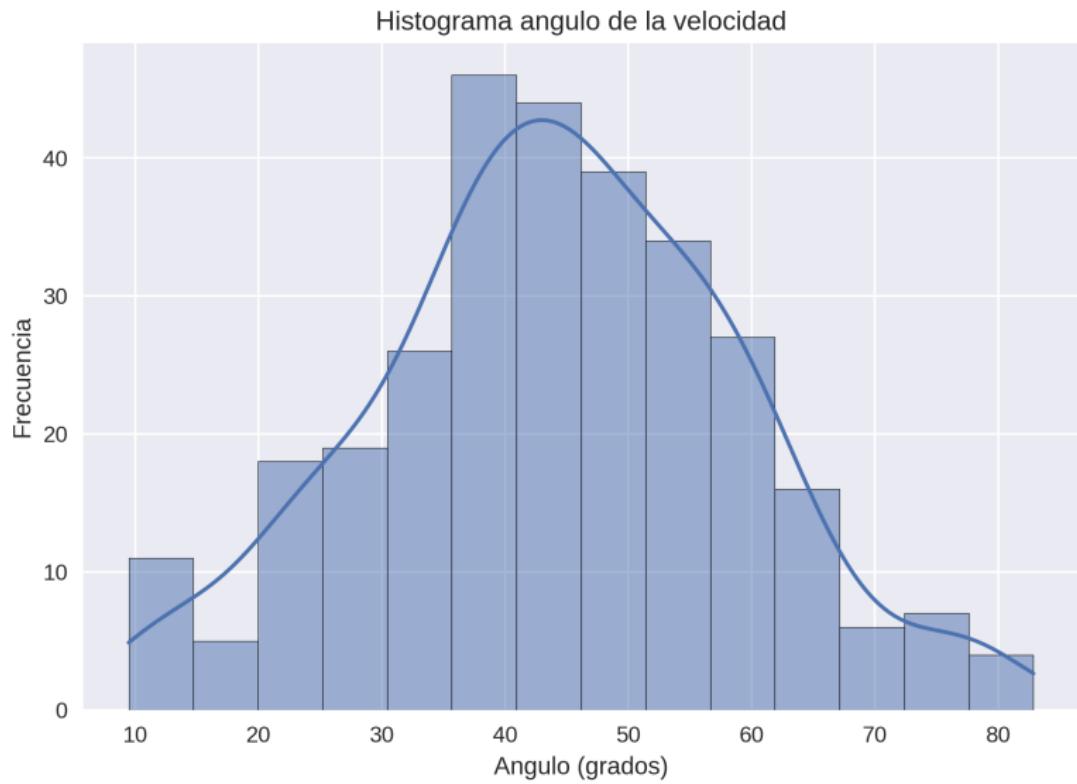
Diagrama de dispersion para velocidades



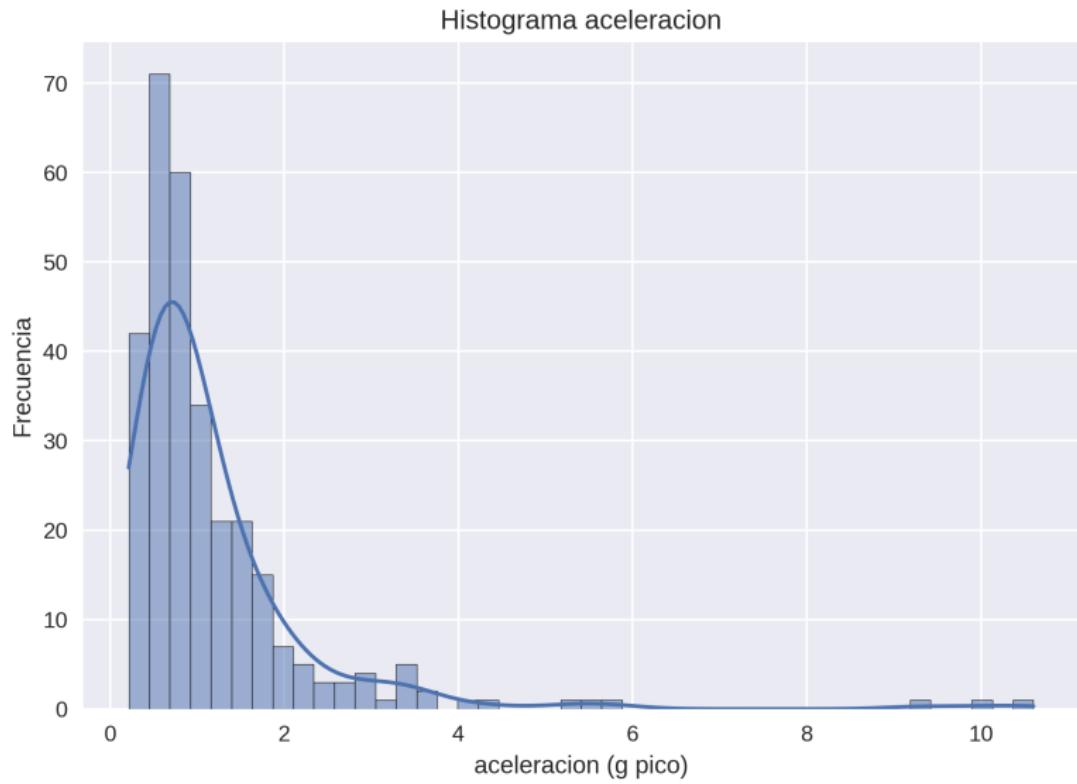
Análisis Exploratorio



Análisis Exploratorio



Análisis Exploratorio



Bases de datos

¿Qué es una Base de datos?

Bases de datos

¿Qué es una Base de datos?

Tipos de Bases de datos.

MongoDB

Clusters

Clusters

Colecciones y Documentos.

MongoDB

Clusters

Colecciones y Documentos.

Json → Bson.

Hosting



Hosting



Colección MotorInDB

Elemento	tipo	Descripción
_id	[]bytes	Elemento utilizado por MongoDB para identificar y facilitar la búsqueda de los documentos
IdMotor	[]string	Lista que contiene todos los identificadores únicos de los motores. Facilita búsqueda e implementación de los clientes Webs

El símbolo “[]” indica que es un arreglo de ese tipo de datos.

Colección MotorData

Elemento	tipo de dato	Descripción
_id	[]bytes	Elemento utilizado por MongoDB para identificar y facilitar la búsqueda de los documentos
IdMotor	string	Identificador único del motor.
Características	string	Descripción e información del motor.
IdSensor	[]uint64	lista de los identificadores-sensores que tiene conectado este motor.
Data	[]DataSensor	lista en forma de sub colección que contiene los resultados del sensor.
Time	int64	Estampa de tiempo, fecha y hora de la muestra en formato Unix.

Colección MotorData

Elemento	tipo de dato	Descripción
UmbralInferiorVel	int32	Umbral Inferior de velocidad.
UmbralSuperiorVel	int32	Umbral Superior de velocidad.
UmbralInferiorAcel	int32	Umbral Inferior de aceleración.
UmbralSuperiorAcel	int32	Umbral Superior de aceleración.

Estructura DataSensor

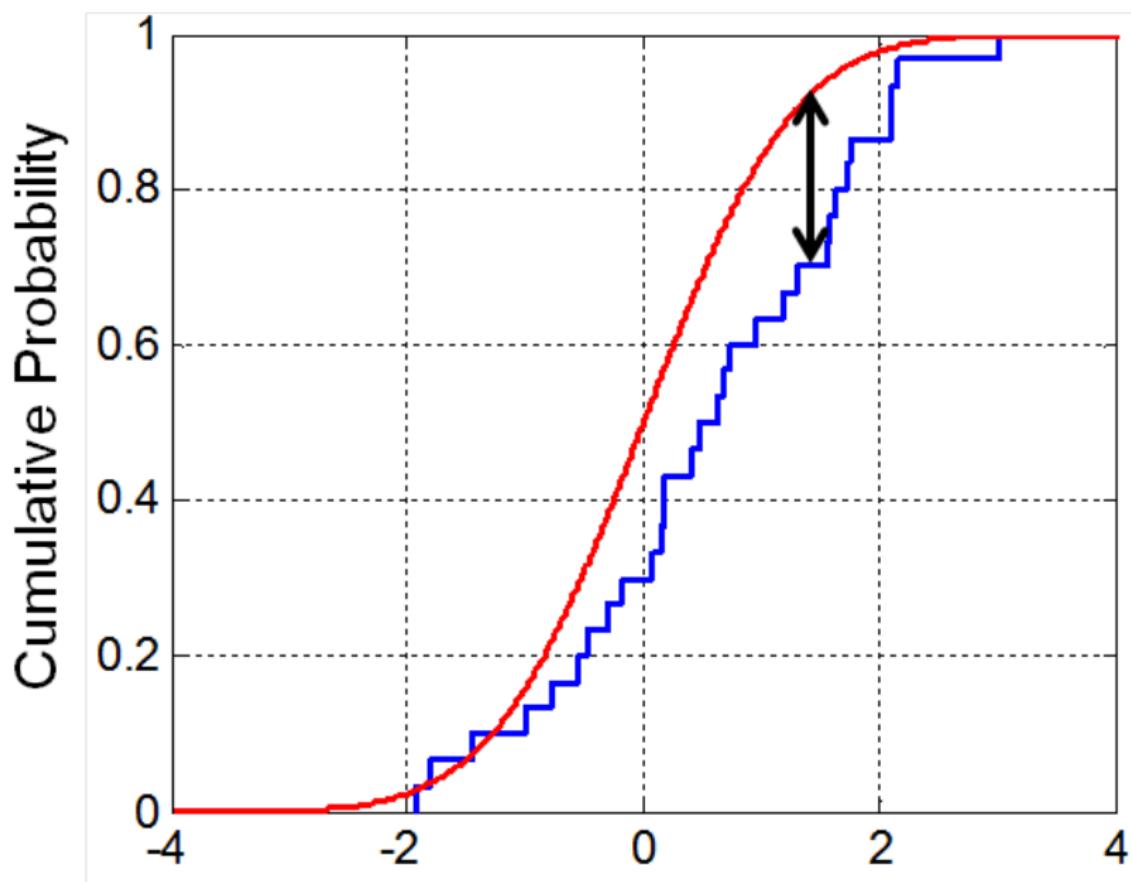
Sub colección “DataSensor”		
IdSensorData	uint64	Identificador único del sensor que tomo la muestra.
Aceleración	float64	Muestra de aceleración medida en g .
VelocidadX	float64	Muestra de velocidad en el eje X.
VelocidadY	float64	Muestra de velocidad en el eje Y.
VelocidadZ	float64	Muestra de velocidad en el eje Z.

Estructura de IdSensor

Tipo de sensor	Reservado	Ubicación	Serial
N_{15}	$N_{14}N_{13}$	N_{12}	$N_{11} \dots N_0$

Campo	Descripción
tipo	tipo de sensor usado, Acelerómetro, Temperatura, etc. Con 0000b siendo Acelerómetro y 15 posibilidades adicionales.
Reservado	No se utilizan, son 0x00 siempre y se reservan para posibles expansiones y/o necesidades.
Ubicación	Posición con respecto al motor y acoplos. Con: 0000b Lado con carga, 0001b Lado libre, 0010b … 1111b disponibles para acoplos y chumaceras.
Serial	número de fabricación del sensor, desde 0 hasta 2^{48} .

Modelo estadísticos



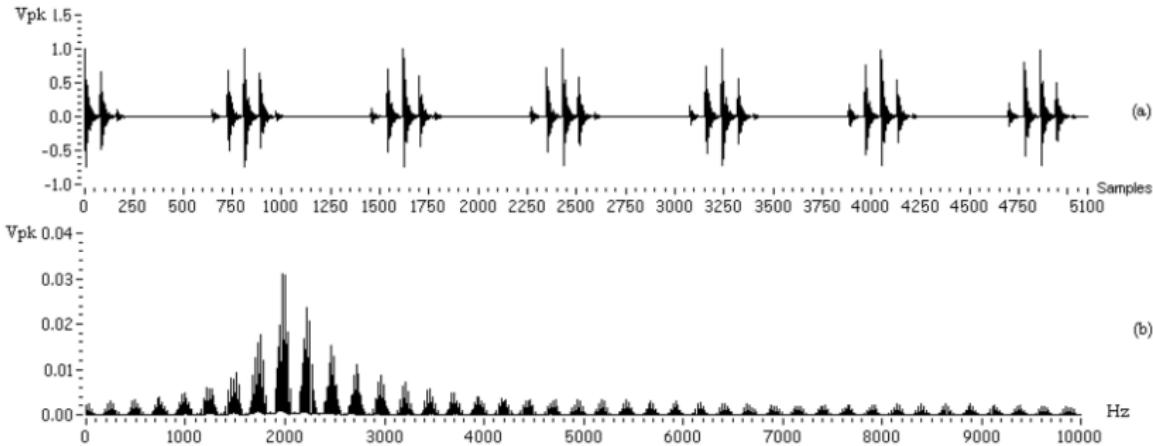
Modelo estadísticos

```
In [3]: def fitting_dist(measurements):
    list_of_dists = ['alpha', 'anglit', 'arcsine', 'beta', 'betaprime',
                    'bradford', 'burr', 'burr12', 'cauchy', 'chi', 'chi2',
                    'cosine', 'dgamma', 'dweibull', 'expon', 'exponnorm',
                    'exponweib', 'exponpow', 'f', 'fatiguelife', 'fisk',
                    'foldcauchy', 'foldnorm', 'genlogistic', 'genpareto',
                    'gennorm', 'genexpon', 'genextreme', 'gausshyper',
                    'gamma', 'gengamma', 'genhalflogistic', 'gilbrat',
                    'gompertz', 'gumbel_r', 'gumbel_l', 'halfcauchy',
                    'halflogistic', 'halfnorm', 'halfgennorm', 'hypsecant',
                    'invgamma', 'invgauss', 'invweibull', 'johnsonsb',
                    'johnsonsu', 'kstwobign', 'laplace', 'levy', 'levy_l',
                    'logistic', 'loggamma', 'loglaplace', 'lognorm',
                    'lomax', 'maxwell', 'mielke', 'nakagami', 'ncx2', 'ncf',
                    'nct', 'norm', 'pareto', 'pearson3', 'powerlaw',
                    'powerlognorm', 'powernorm', 'rdist', 'reciprocal',
                    'rayleigh', 'rice', 'recipinvgauss', 'semicircular',
                    't', 'triang', 'truncrexpon', 'truncnorm', 'tukeylambda',
                    'uniform', 'vonmises', 'vonmises_line', 'wald',
                    'weibull_min', 'weibull_max']
    results = []
    for dist_name in list_of_dists:
        dist = getattr(stats, dist_name)
        param = dist.fit(measurements)
        _, pval = stats.kstest(measurements, dist_name, args=param)
        results.append((pval, dist_name))
    results.sort(reverse=True)
    return filter(lambda x: x > (0.05,), results)
```

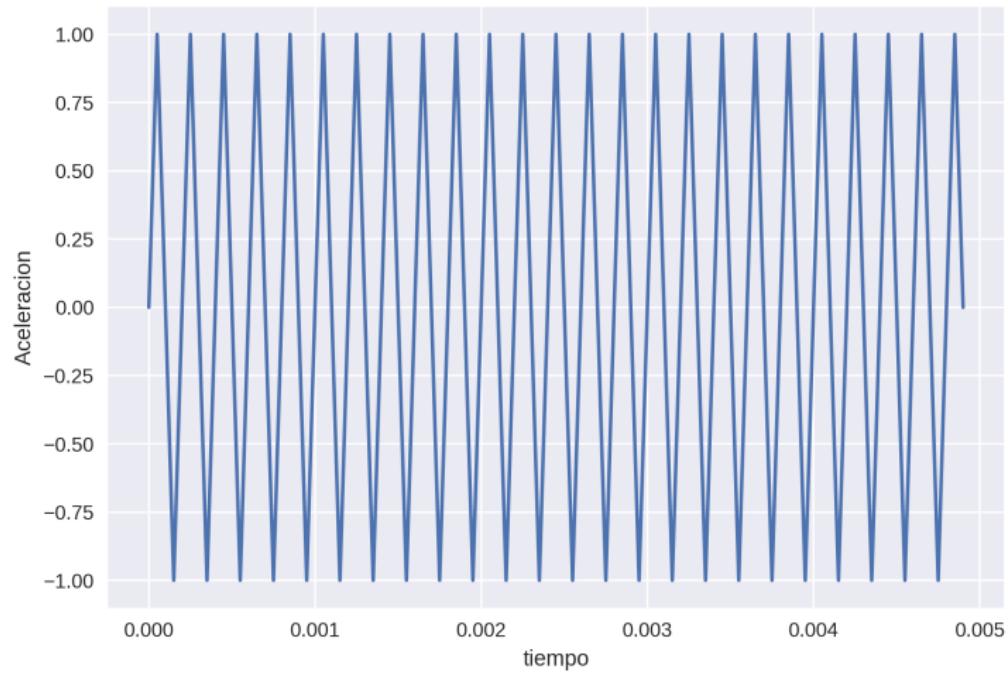
Modelo estadísticos



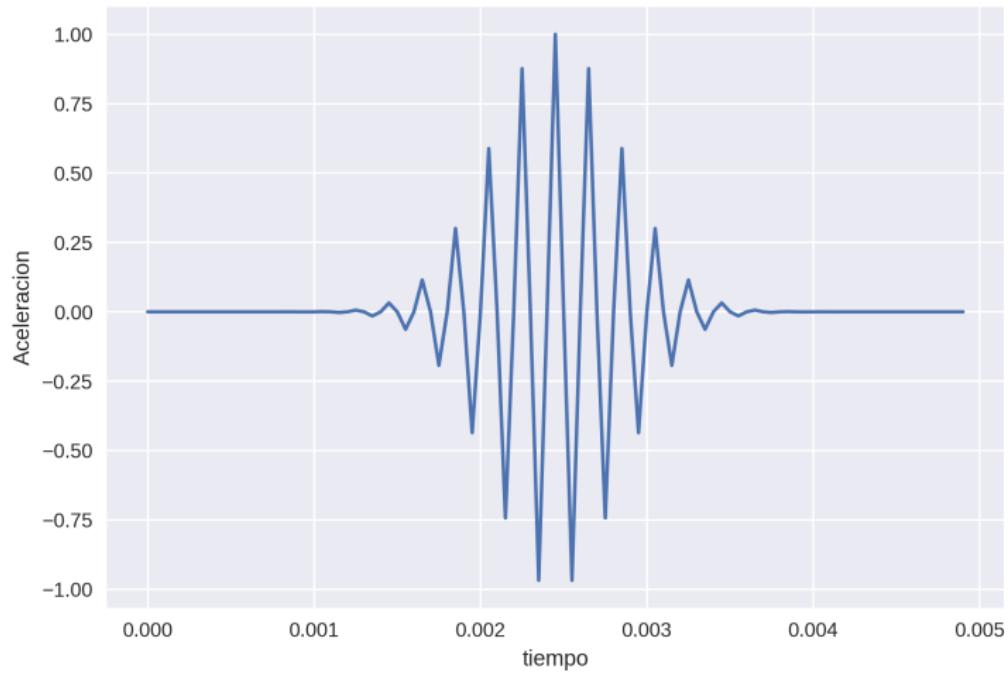
Modelo estadísticos



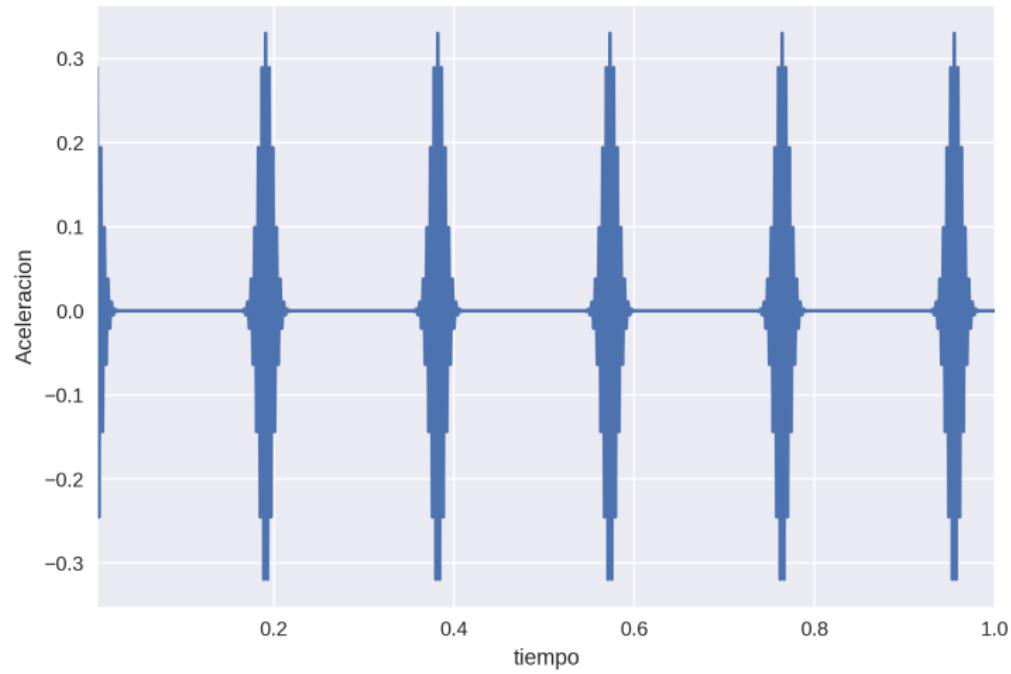
Modelo estadísticos



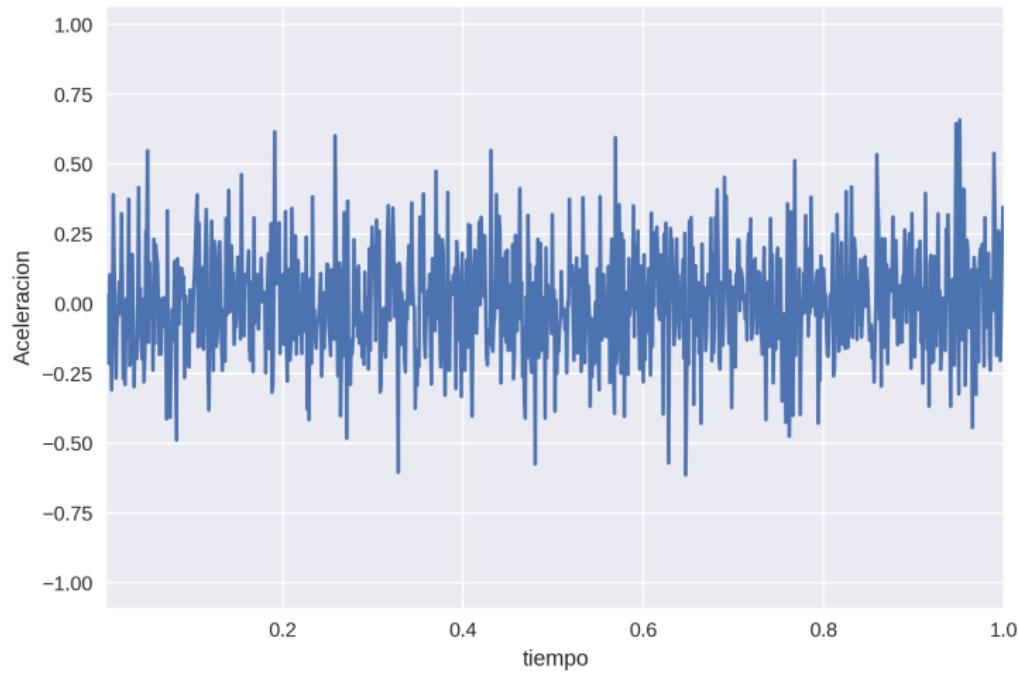
Modelo estadísticos



Modelo estadísticos



Modelo estadísticos





DigitalOcean

¿Qué son las API?

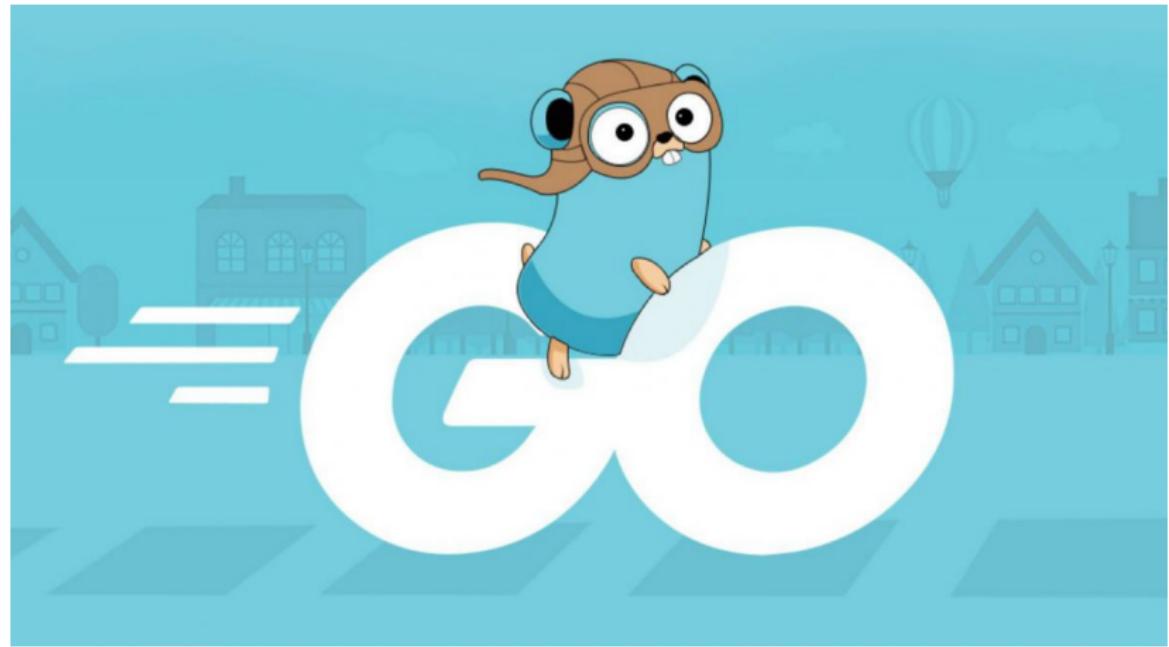
API

¿Qué son las API?

¿Qué ventajas ofrecen?



Golang



End Points

Dirección	Tarea realizada
DireccionIP/sensormessage	<p>controlar el comportamiento, acceso e intercambio de información sensor-servidor.</p> <p>Es una comunicación bidireccional con Http2 (Https) y se intercambian por el canal establecido al comenzar la conexión.</p>
DireccionIP/exhaustive ?idMotor=identificador	Solicitar la información para la visita exhaustiva, el identificador único del motor que se solicita la data es enviado por el Url (?idMotor=identificador) .

Estructura medición exhaustiva

Elemento	tipo de dato	Descripción
IdMotor	string	Identificador único del motor.
Características	string	Descripción e información del motor.
IdSensor	[]uint64	lista de los identificadores-sensores que tiene conectado este motor.
Data	[]DataExhaustive	lista en forma de sub colección que contiene los resultados del sensor.
Time	int64	Estampa de tiempo, fecha y hora de la muestra en formato Unix.

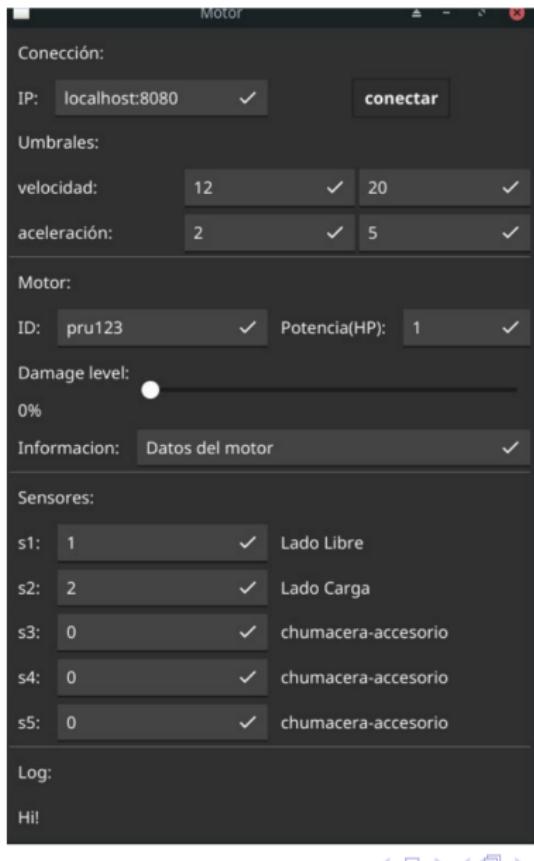
El símbolo “[]” indica que es un arreglo de ese tipo de datos.

Estructura DataExhaustiva

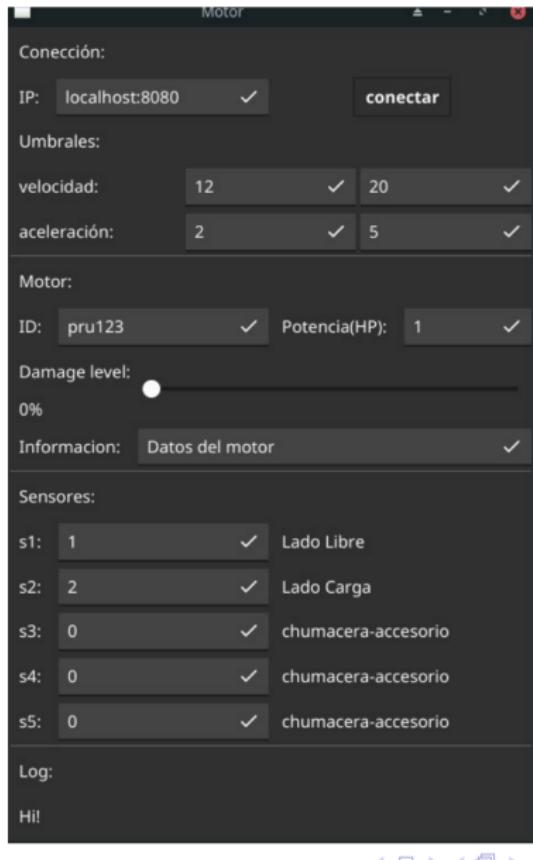
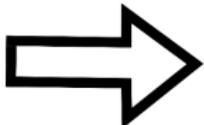
estructura “DataExhaustiva”		
IdSensorData	uint64	Identificador único del sensor que tomo la muestra.
Data	[]float64	Lista de mediciones de aceleración, medidas en g.

El símbolo “[]” indica que es un arreglo de ese tipo de datos.

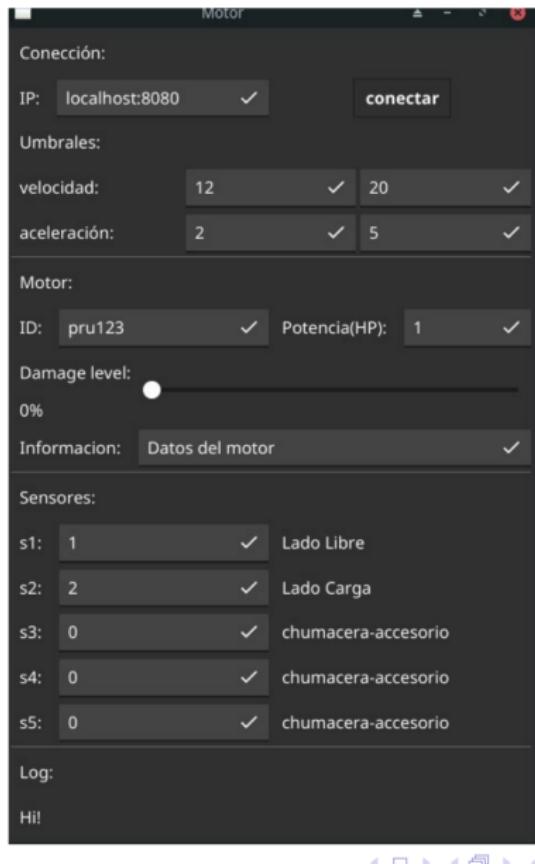
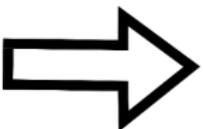
Cliente Sensorica



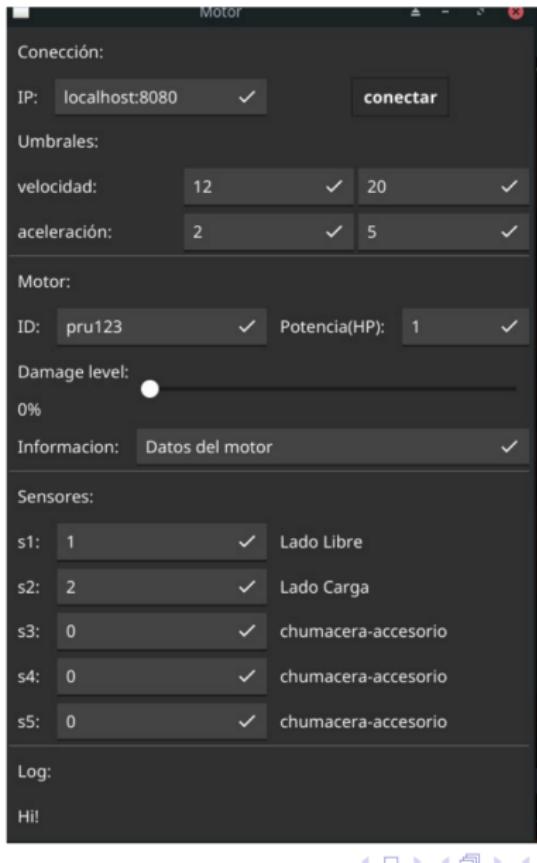
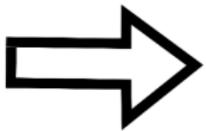
Cliente Sensorica



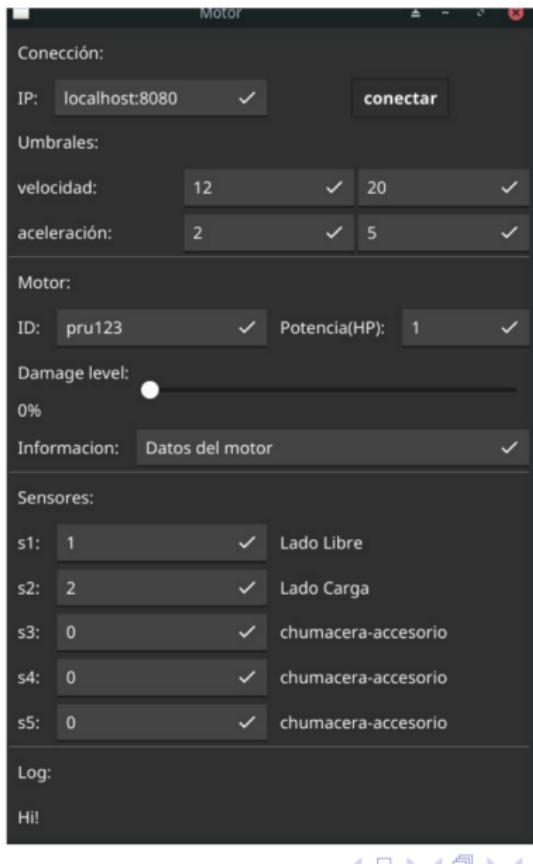
Cliente Sensorica



Cliente Sensorica

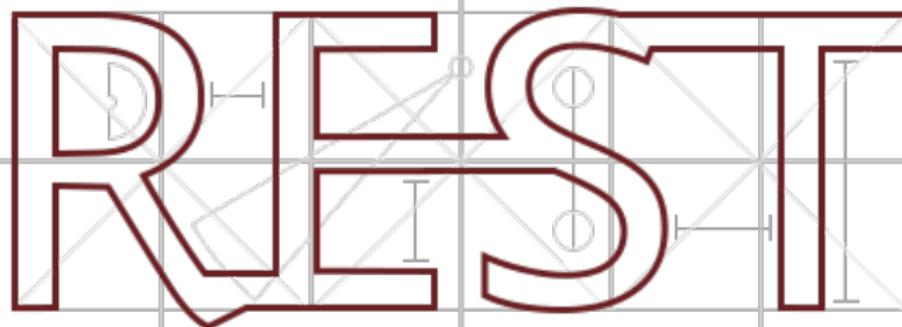


Cliente Sensorica



django

django



REST

framework

Servidor Web

Dirección	Tarea realizada
DireccionIP/	Enviar los archivos necesarios para mostrar la vista general en el navegador.
DireccionIP/especifica Motor=identificador	?Id- Enviar los archivos necesarios para mostrar la vista específica en el navegador.
DireccionIP/exhaustiva Motor=identificador	?Id- Enviar los archivos necesarios para mostrar la vista exhaustiva en el navegador.

Servidor Web

Dirección	Tarea realizada
DireccionIP/ static/...	Envío de archivos estáticos por solicitudes externas, ya sea por un requerimiento del HTML o del JavaScript.
DireccionIP/ get_data_motores	API que entrega una lista, formato JSON, con la ultima medición almacenada de cada sensor en la base de datos.
DireccionIP/ get_list_motores	API que entrega una lista, formato JSON, con todos los motores de los que se dispone información en la base de datos.

Servidor Web

Dirección	Tarea realizada
DireccionIP/get_especifica ?IdMotor=identificador	API que se encarga de devolver un JSON con toda la información en la base de datos referente al motor con el Id pasado en el Url, además de las direcciones para las gráficas históricas
DireccionIP/get_exhaustiva ?IdMotor=identificador	API que se encarga de devolver un JSON con toda la información en la base de datos referente al motor con el Id pasado en el Url, además de las direcciones para las gráficas históricas y exhaustiva.
DireccionIP/help	Endpoint que se encarga de enviar el HTML referente al manual de usuario.

Cliente web

HTML

Cliente web

HTML

CSS

Cliente web

HTML

CSS

JavaScript → ReactJs.

Cliente Web, vista general

Planta x...

Aqui va a ir informacion

BUSCAR Seleccion un Motor

RESET - 1 +

 Motor Bien1	 Motor Medio2	 Motor Medio3	 Motor Mal4
 Motor Nivel1	 Motor Nivel2	 Motor Nivel3	 Motor Frontera1
 Motor Frontera2	 Motor Graficas1	 Motor Prueba1	

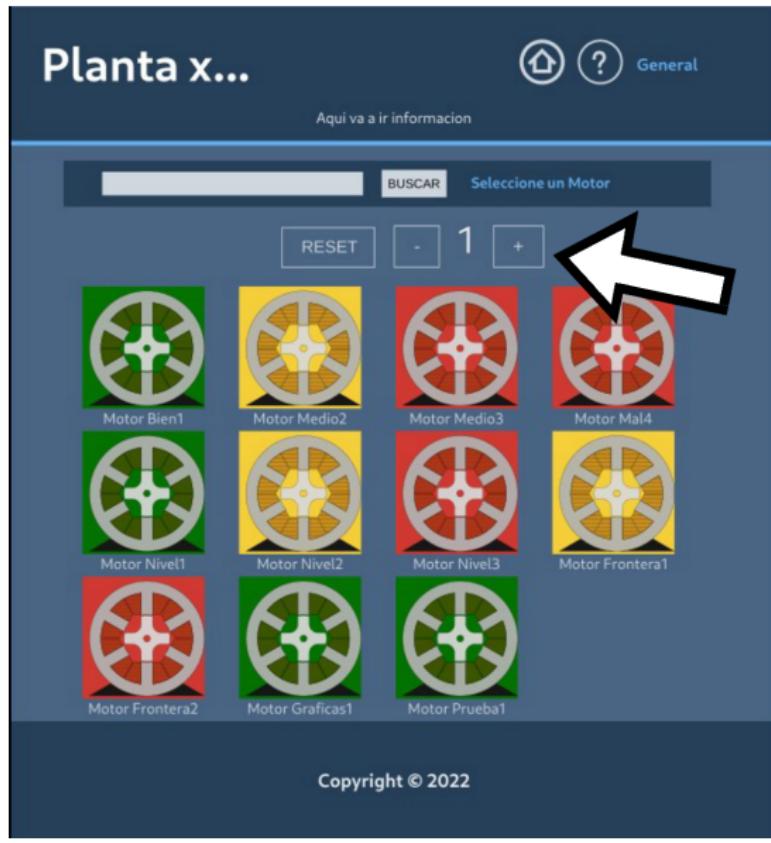
Copyright © 2022



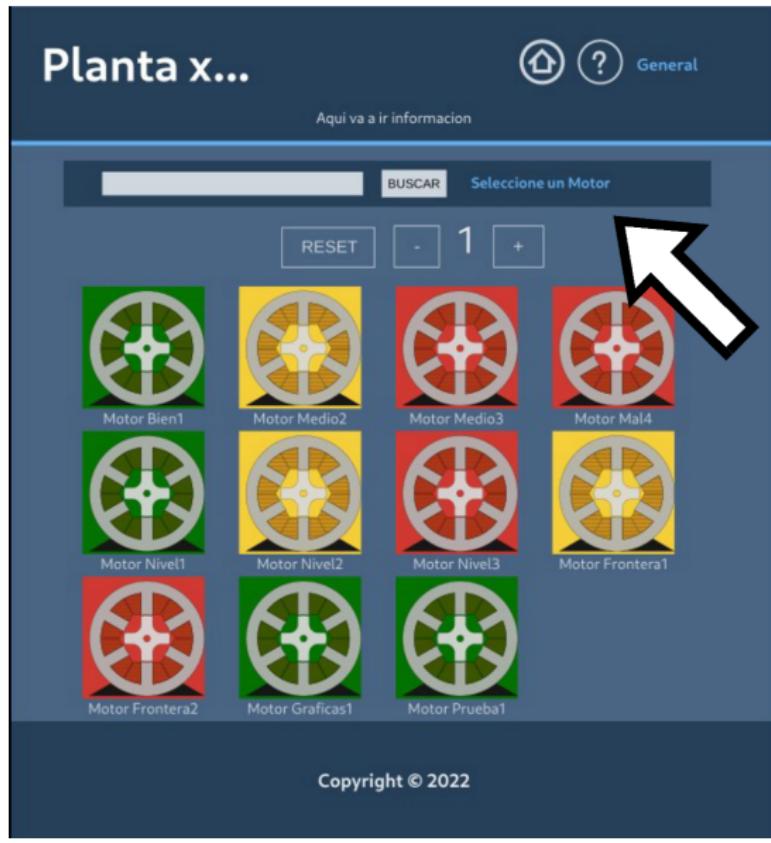
Cliente Web, vista general



Cliente Web, vista general



Cliente Web, vista general



Cliente Web, vista específica

Planta x...

Aqui va a ir informacion

Estado del Motor:

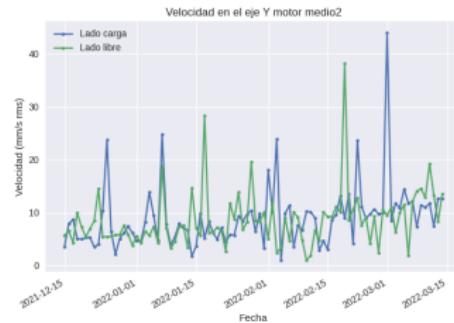
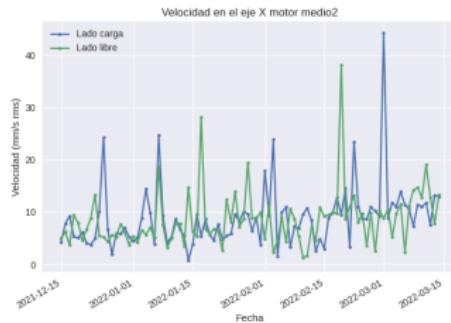
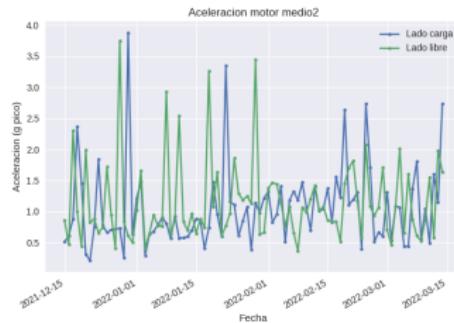
caracteristicas y comentarios ejemplo, buen estado

Histogramas:

 Motor Nivel1

Solicitar Vista Exhaustiva

Cliente Web, vista específica



Cliente Web, vista específica

Tabla resumen:

HISTOGRAMA COMO EXCEL

FECHA	ID SENSOR	LADO	VELOCIDAD VERTICAL	VELOCIDAD HORIZONTAL	VELOCIDAD AXIAL
14/12/2021	15	Carga	5.330071390672369	6.514656414273091	2.5000000000000002
14/12/2021	7b	Libre	7.5652013532957	3.686068758829853	2.5000000000000002

Copyright © 2022

Cliente Web, vista exhaustiva

Planta x... Exhaustiva

Aqui va a ir informacion

Estado del Motor:

caracteristicas y comentarios ejemplo, buen estado

Vista en frecuencia:

Histogramas:

Cliente Web, vista exhaustiva

Planta x... Exhaustiva

Aqui va a ir informacion

Estado del Motor:

Regresar A Vista General

Nivel1

caracteristicas y comentarios ejemplo, buen estado

Vista en frecuencia:

Histogramas:

Cliente Web, vista exhaustiva



Cliente Web, vista exhaustiva

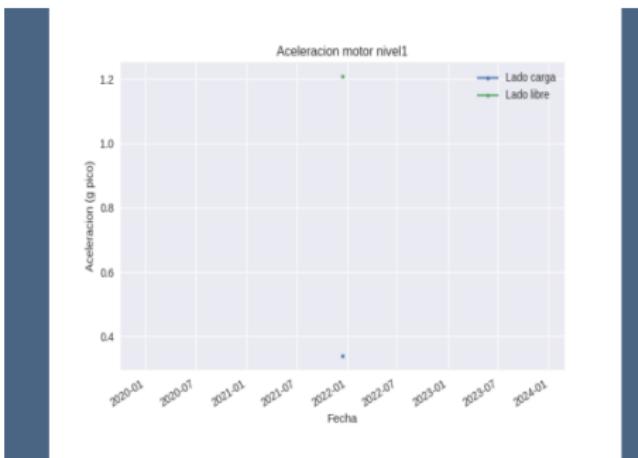


Tabla resumen:

HISTOGRAMA COMO EXCEL

FECHA	ID SENSOR	LADO	VELOCIDAD VERTICAL	VELOCIDAD HORIZONTAL	VELOCIDAD AXIAL
14/12/2021	15	Carga	5.330071390672369	6.514656414273091	1.2000000000000002
14/12/2021	7b	Libre	7.5652013532957	3.686068758829853	1.2000000000000002

Copyright © 2022

Scripts

¿Qué son?

Scripts

¿Qué son?

¿En que lenguajes se pueden implementar?

Llenado de la base de datos

```
Terminal - gerardo@gerardo-intelpoweredclassmatepc:~/tesis-go/scripts
File Edit View Terminal Tabs Help
[gerardo@gerardo-intelpoweredclassmatepc scripts]$ ./LlenarBBDD -i muestral -s1 234 -s2 1454 -s3 56
6 -c "Utilizando el script para llenar la BBDD" -d 3
2022/04/04 18:00:54 Conexion exitosa con la BD
2022/04/04 18:00:54 agregado a la BBDD
2022/04/04 18:00:55 El id de la impresion es
2022/04/04 18:00:55 ObjectId("624b6a977b5167a1f823de26")
2022/04/04 18:00:56 El id de la impresion es
2022/04/04 18:00:56 ObjectId("624b6a987b5167a1f823de27")
2022/04/04 18:00:57 El id de la impresion es
2022/04/04 18:00:57 ObjectId("624b6a987b5167a1f823de28")
2022/04/04 18:00:57 El id de la impresion es
2022/04/04 18:00:57 ObjectId("624b6a997b5167a1f823de29")
2022/04/04 18:00:58 El id de la impresion es
2022/04/04 18:00:58 ObjectId("624b6a9a7b5167a1f823de2a")
2022/04/04 18:00:59 El id de la impresion es
2022/04/04 18:00:59 ObjectId("624b6a9b7b5167a1f823de2b")
2022/04/04 18:00:59 El id de la impresion es
2022/04/04 18:00:59 ObjectId("624b6a9b7b5167a1f823de2c")
2022/04/04 18:01:00 El id de la impresion es
2022/04/04 18:01:00 ObjectId("624b6a9c7b5167a1f823de2d")
2022/04/04 18:01:01 El id de la impresion es
2022/04/04 18:01:01 ObjectId("624b6a9c7b5167a1f823de2e")
2022/04/04 18:01:01 El id de la impresion es
2022/04/04 18:01:01 ObjectId("624b6a9d7b5167a1f823de2f")
2022/04/04 18:01:02 El id de la impresion es
2022/04/04 18:01:02 ObjectId("624b6a9e7b5167a1f823de30")
2022/04/04 18:01:02 El id de la impresion es
2022/04/04 18:01:02 ObjectId("624b6a9e7b5167a1f823de31")
2022/04/04 18:01:03 El id de la impresion es
2022/04/04 18:01:03 ObjectId("624b6a9f7b5167a1f823de32")
```

Sensor

```
[gerardo@gerardo-intelpoweredclassmatepc scripts]$ ./clienteScript -i probandoScript1 -s1 525 -s2 4  
54 -uiv 4 -uia 1 -usv 11 -usa 8 -dam 2 -c "probando todas las banderas"  
2022/04/01 16:50:00 0  
.....-2022/04/01 16:50:06 enviado post  
.....-2022/04/01 16:50:11 enviado post  
.....-2022/04/01 16:50:16 enviado post  
.....-2022/04/01 16:50:21 enviado post  
.....-2022/04/01 16:50:26 enviado post  
.....-2022/04/01 16:50:31 enviado post  
.....-2022/04/01 16:50:36 enviado post  
.....-2022/04/01 16:50:41 enviado post  
.....-2022/04/01 16:50:46 enviado post  
2022/04/01 16:50:46 exhaustiva  
2022/04/01 16:50:46 Listen send  
2022/04/01 16:50:46 1  
2022/04/01 16:50:46 2  
2022/04/01 16:50:47 enviado exhaustiva  
.....-2022/04/01 16:50:51 enviado post  
.....-2022/04/01 16:50:56 enviado post  
.....-2022/04/01 16:51:01 enviado post
```

Comprobación de resultados

Comprobación técnica

/lib/systemd/system/goweb.service

```
[Unit]
Description=goweb

[Service]
Type=simple
Restart=always
RestartSec=5s
ExecStart=/home/user/go/go-web/main

[Install]
WantedBy=multi-user.target
```

Comprobación técnica

The image shows three terminal windows side-by-side. The leftmost window displays the execution of a Go program named 'cliente.go'. It shows the server listening on port 8080 and receiving a connection from the client. The middle window shows the execution of another Go program, likely 'main.go', which connects to a database and performs a query. The rightmost window shows a failed login attempt, resulting in an exit status of 1.

```
[gerardo@gerardo-intelpoweredclassmatepc ~]$ go run cliente.go
2022/03/06 16:10:05 localhost:8080 Datos del motor
2022/03/06 16:10:05 2451
2022/03/06 16:10:05 1
2022/03/06 16:10:05 0
2022/03/06 16:10:05
valor:0 1
2022/03/06 16:10:05
valor:1 1000000000002
[gerardo@gerardo-intelpoweredclassmatepc ~]$ 

[gerardo@gerardo-intelpoweredclassmatepc tesis-go]$ g
o run cliente.go
2022/03/06 16:10:09 localhost:8080 Datos del motor
2022/03/06 16:10:09 2451
2022/03/06 16:10:09 1
2022/03/06 16:10:09 0
2022/03/06 16:10:09
valor:0 1
2022/03/06 16:10:09
valor:1 1000000000002
2022/03/06 16:10:09 Failed login 2: <nil>
exit status 1
[gerardo@gerardo-intelpoweredclassmatepc tesis-go]$ 

[gerardo@gerardo-intelpoweredclassmatepc ~]$ cd tesis-go
[gerardo@gerardo-intelpoweredclassmatepc ~]$ go run main.go
2022/03/06 16:08:53 Conexion exitosa con la BD
2022/03/06 16:08:53 comenzo
2022/03/06 16:08:53 Serving on 8080
2022/03/06 16:10:06 [::1]:53620] Trata de conectarse: 2451
2022/03/06 16:10:06 login sucesed
2022/03/06 16:10:06 agregado a la BD
2022/03/06 16:10:06 El id de la impresion es
2022/03/06 16:10:06 ObjectId("6225151e8aee853f87604a42")
2022/03/06 16:10:09 [::1]:53622] Trata de conectarse: 2451
2022/03/06 16:10:09 http2: panic serving [::1]:53622: runtime error: invalid memory address or
nil pointer dereference
goroutine 209 [running]:
net/http.(*httpServerConn).runHandler.func1()
/usr/lib/go/src/net/http/h2_bundle.go:5842 +0x125
```

Comprobación técnica

The screenshot shows a terminal window with the following details:

- Title bar:** Terminal - gerardo@gerardo-intelpoweredclassmatepc:~/tesis-go
- Menu bar:** File Edit View Terminal Tabs Help
- Command line:** [gerardo@gerardo-intelpoweredclassmatepc tesis-go]\$ go run main.go
- Output:**

```
2022/03/06 15:55:04 error parsing uri: lookup _mongodb._tcp.cluster0.aohmy.mongo
db.net on [::1]:53: read udp [::1]:54546->[::1]:53: read: connection refused
exit status 1
```
- Bottom right corner:** A small icon representing a terminal or command line interface.

Comprobación tecnica

```
File "/home/andres/.local/share/virtualenvs/tesis-python-XUoJKyxP/lib/python3.10/site-packages/pymongo/mongo_client.py", line 1553, in __start_session
    server_session = self._get_server_session()
    session_timeout = self._check_session_support()
File "/home/andres/.local/share/virtualenvs/tesis-python-XUoJKyxP/lib/python3.10/site-packages/pymongo/mongo_client.py", line 1589, in _get_server_session
    return self._topology.get_server_session()
File "/home/andres/.local/share/virtualenvs/tesis-python-XUoJKyxP/lib/python3.10/site-packages/pymongo/topology.py", line 530, in get_server_session
    session_timeout = self._check_session_support()
File "/home/andres/.local/share/virtualenvs/tesis-python-XUoJKyxP/lib/python3.10/site-packages/pymongo/topology.py", line 514, in _check_session_support
    self._select_servers_loop()
File "/home/andres/.local/share/virtualenvs/tesis-python-XUoJKyxP/lib/python3.10/site-packages/pymongo/topology.py", line 216, in _select_servers_loop
    raise ServerSelectionTimeoutError(
pymongo.errors.ServerSelectionTimeoutError: cluster0-shard-00-01.aohmy.mongodb.net:27017: [Errno -3] Temporary failure in name resolution,cluster0-shard-00-02.aohmy.mongodb.net:27017: [Errno -3] Temporary failure in name resolution,cluster0-shard-00-00.aohmy.mongodb.net:27017: [Errno -3] Temporary failure in name resolution, Timeout: 30s, Topology Description: <TopologyDescription id: 62250203beba41459b45f4a4, topology_type: ReplicaSetNoPrimary, servers: [<ServerDescription ('cluster0-shard-00-00.aohmy.mongodb.net', 27017) server_type: Unknown, rtt: None, error=AutoReconnect('cluster0-shard-00-00.aohmy.mongodb.net:27017: [Errno -3] Temporary failure in name resolution')>, <ServerDescription ('cluster0-shard-00-01.aohmy.mongodb.net', 27017) server_type: Unknown, rtt: None, error=AutoReconnect('cluster0-shard-00-01.aohmy.mongodb.net:27017: [Errno -3] Temporary failure in name resolution')>, <ServerDescription ('cluster0-shard-00-02.aohmy.mongodb.net', 27017) server_type: Unknown, rtt: None, error=AutoReconnect('cluster0-shard-00-02.aohmy.mongodb.net:27017: [Errno -3] Temporary failure in name resolution')>]
[06/Mar/2022 18:49:09] "GET /get_list_motores HTTP/1.1" 500 141696
[06/Mar/2022 18:49:20] "GET /get_data_motores HTTP/1.1" 200 8727
```

Comprobación tecnica

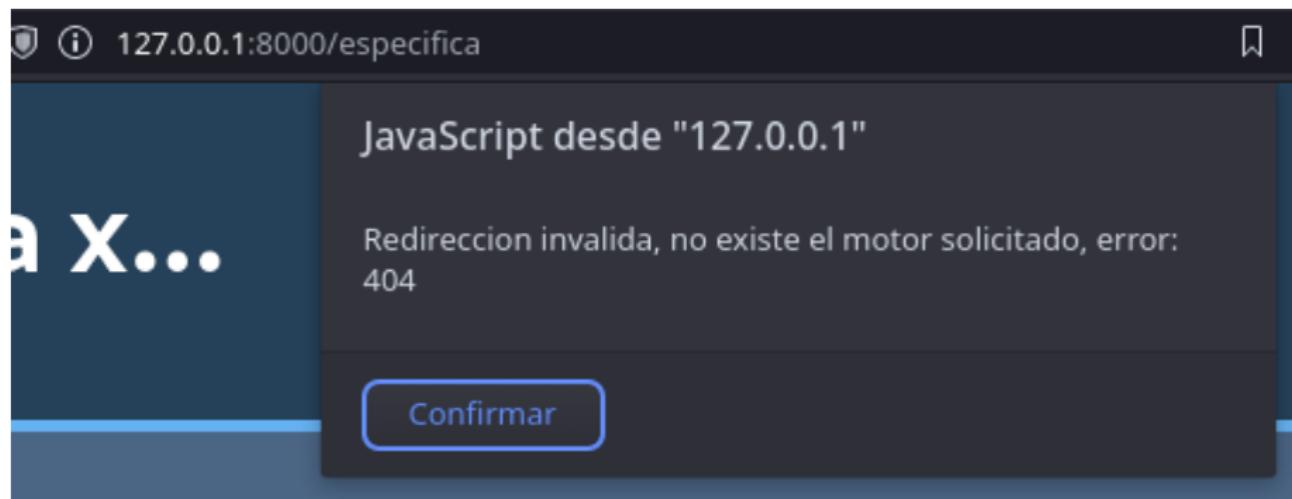


The screenshot shows a web browser window with the URL `https://localhost:8080/exhaustive`. The status bar indicates a failed connection with a red lock icon.

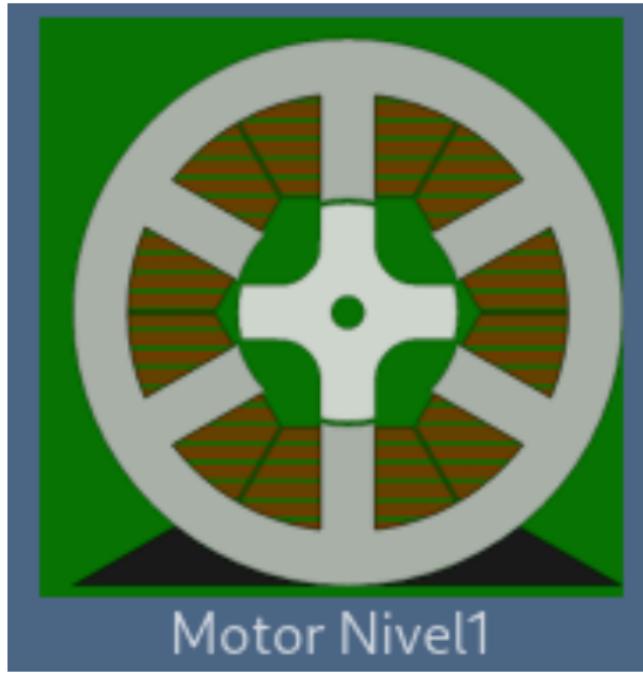
```
Terminal - gerardo@gerardo-intelpoweredclassmatepc:~/tesis-go
File Edit View Terminal Tabs Help
[gerardo@gerardo-intelpoweredclassmatepc tesis-go]$ go run main.go
2022/03/06 15:51:20 Conexion exitosa con la BD
2022/03/06 15:51:20 comenzo
2022/03/06 15:51:20 Serving on 8080
2022/03/06 15:52:13 http: TLS handshake error from [::1]:53612: remote error: tls: unknown certificate
```

The terminal window displays the output of the `go run main.go` command. It shows successful database connection, start, and serving on port 8080, followed by a TLS handshake error due to an unknown certificate.

Comprobación técnica

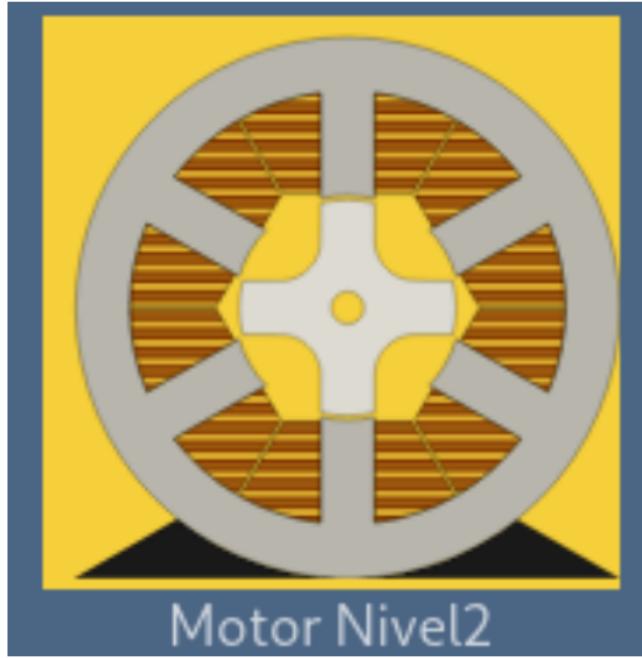


Comprobación de resultados



```
▼ Data: Array
  ▼ 0: Object
    IdSens... : 21
    Aceler... : 0.33945888467322044
    Veloci... : 6.514656414273091
    Veloci... : 5.330071390672369
    Veloci... : 3
  ▼ 1: Object
    IdSens... : 281474976710779
    Aceler... : 1.208568996642277
    Veloci... : 3.686068758829853
    Veloci... : 7.5652013532957
    Veloci... : 2
```

Comprobación de resultados



```
✓ Data: Array
  ✓ 0: Object
    IdSens... : 21
    Aceler... : 2.3394588846732205
    Veloci... : 16.51465641427309
    Veloci... : 15.33007139067237
    Veloci... : 13
  ✓ 1: Object
    IdSens... : 281474976710779
    Aceler... : 2.208568996642277
    Veloci... : 13.686068758829853
    Veloci... : 17.5652013532957
    Veloci... : 12.2
```

Comprobación de resultados



```
✓ Data: Array
  ✓ 0: Object
    IdSens... : 21
    Aceler... : 6.3394588846732205
    Veloci... : 26.51465641427309
    Veloci... : 25.33007139067237
    Veloci... : 23
  ✓ 1: Object
    IdSens... : 281474976710779
    Aceler... : 5.208568996642277
    Veloci... : 22.686068758829855
    Veloci... : 27.5652013532957
    Veloci... : 22
```

Comprobación de resultados



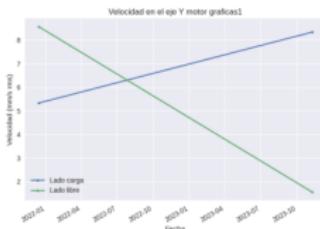
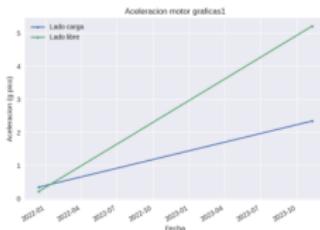
```
✓ Data: Array
  ✓ 0: Object
    IdSens... : 21
    Aceler... : 2
    Veloci... : 12
    Veloci... : 12
    Veloci... : 12
  ✓ 1: Object
    IdSens... : 281474976710779
    Aceler... : 2
    Veloci... : 12
    Veloci... : 12
    Veloci... : 12
```

Comprobación de resultados



```
✓ Data: Array
  ✓ 0: Object
    IdSens... : 21
    Aceler... : 5
    Veloci... : 20
    Veloci... : 20
    Veloci... : 20
  ✓ 1: Object
    IdSens... : 281474976710779
    Aceler... : 5
    Veloci... : 20
    Veloci... : 20
    Veloci... : 20
```

Comprobación de resultados



Comprobación de resultados

<pre>IdMotor: "graficas1" caracte...: "lineas de graficas" > IdSensor: Array ✓ Data: Array ✓ 0: Object IdSens...: 21 Aceler...: 0.33945888467322044 Veloci...: 6.514656414273091 Veloci...: 5.330071390672369 Veloci...: 0 ✓ 1: Object IdSens...: 281474976710779 Aceler...: 0.208568996642277 Veloci...: 10.686068758829853 Veloci...: 8.5652013532957 Veloci...: 0 Time: 1639518818</pre>	<pre>IdMotor: "graficas1" caracte...: "lineas de graficas" > IdSensor: Array ✓ Data: Array ✓ 0: Object IdSens...: 21 Aceler...: 2.3394588846732205 Veloci...: 5.514656414273091 Veloci...: 8.33007139067237 Veloci...: 5 ✓ 1: Object IdSens...: 281474976710779 Aceler...: 5.208568996642277 Veloci...: 1.686068758829853 Veloci...: 1.5652013532957 Veloci...: 9 Time: 169951RR1R</pre>																																			
<p>Tabla resumen:</p> <p>HISTOGRAFICO COMO EXCEL</p> <table border="1"><thead><tr><th>FECHA</th><th>ID SENSOR</th><th>LADO</th><th>VELOCIDAD VERTICAL</th><th>VELOCIDAD HORIZONTAL</th><th>VELOCIDAD AERIAL</th><th>ACELERACION</th></tr></thead><tbody><tr><td>14/12/2021</td><td>15</td><td>Carga</td><td>5.330071390672369</td><td>6.514656414273091</td><td>0</td><td>0.3394588846732044</td></tr><tr><td>14/12/2021</td><td>76</td><td>Libre</td><td>8.5652013532957</td><td>10.686068758829853</td><td>0</td><td>5.208568996642277</td></tr><tr><td>07/11/2023</td><td>15</td><td>Carga</td><td>8.33007139067237</td><td>5.514656414273091</td><td>5</td><td>2.3394588846732105</td></tr><tr><td>07/11/2023</td><td>76</td><td>Libre</td><td>1.5652013532957</td><td>1.686068758829853</td><td>9</td><td>5.208568996642277</td></tr></tbody></table>		FECHA	ID SENSOR	LADO	VELOCIDAD VERTICAL	VELOCIDAD HORIZONTAL	VELOCIDAD AERIAL	ACELERACION	14/12/2021	15	Carga	5.330071390672369	6.514656414273091	0	0.3394588846732044	14/12/2021	76	Libre	8.5652013532957	10.686068758829853	0	5.208568996642277	07/11/2023	15	Carga	8.33007139067237	5.514656414273091	5	2.3394588846732105	07/11/2023	76	Libre	1.5652013532957	1.686068758829853	9	5.208568996642277
FECHA	ID SENSOR	LADO	VELOCIDAD VERTICAL	VELOCIDAD HORIZONTAL	VELOCIDAD AERIAL	ACELERACION																														
14/12/2021	15	Carga	5.330071390672369	6.514656414273091	0	0.3394588846732044																														
14/12/2021	76	Libre	8.5652013532957	10.686068758829853	0	5.208568996642277																														
07/11/2023	15	Carga	8.33007139067237	5.514656414273091	5	2.3394588846732105																														
07/11/2023	76	Libre	1.5652013532957	1.686068758829853	9	5.208568996642277																														

Comprobación de resultados

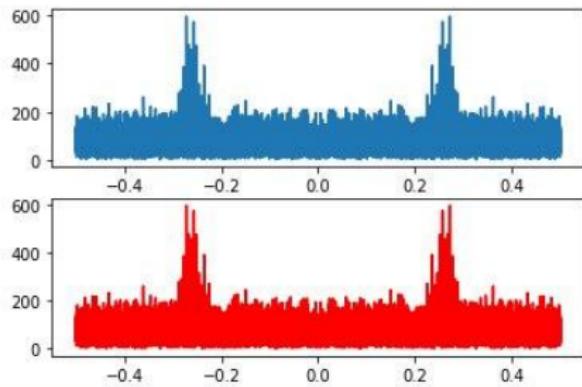
```
In [33]: time_step = 1
freqs = np.fft.fftfreq(xBper.size, time_step)
idx   = np.argsort(freqs)

ps = np.abs(octave.fft(xBper))[0] # Usando la FFT de Octave
plt.subplot(211)
plt.plot(freqs[idx], ps[idx])

plt.subplot(212)
ps = np.abs(np.fft.fft(xBper)) # Usando la FFT de Numpy
plt.plot(freqs[idx], ps[idx],c='r')
```

20000

Out[33]: [`<matplotlib.lines.Line2D at 0x7f8c9e7b42e0>`]



Conclusiones

El desarrollo de una herramienta computacional para el análisis de la vibración en motores eléctricos se hizo posible mediante la implementación de dos servidores, uno de estilo microservicio y un cliente Web, puede ser alimentada mediante la simulación de un modelo estadístico o mediante mediciones a tiempo real de cualquier sistema que respete las estructuras de datos empleado.

Conclusiones

Se implementaron múltiples servicios, cada uno escrito con un lenguaje de programación que satisface los requerimientos de velocidad, paralelismo y robustez necesarios para los requerimientos de uso, además de una fácil adecuación y modificación a las necesidades, de múltiples clientes.

Conclusiones

Existe una gran cantidad de lenguajes de programación en la actualidad diseñados para satisfacer una gran cantidad de tareas en múltiples campos, siendo Go y Python herramientas de gran poder y fácil implementación al realizar servidores y API, además de la creación de Script para la automatización de procesos; de igual forma Python es increíblemente polifacético y cuenta con una gran cantidad de librerías y Frameworks especialmente en el área de la ciencia de datos y estadística.

Conclusiones

En términos de programación Web a nivel cliente, HTML, CSS, JavaScript, son casi una obligación, sin embargo, JavaScript cuenta con una variedad de Frameworks para agilizar el trabajo, siendo React uno de los mas utilizados en la industria.

Conclusiones

Se implementaron dos modelos estadísticos, uno para entregar el equivalente a mediciones diarias y otro para mediciones continuas a tiempo real necesarias para hacer un análisis en frecuencia, utilizando distribuciones de probabilidad para representar 3 variables de interés, velocidad vertical, horizontal y aceleración, expresadas de forma independiente como magnitud y ángulo de la velocidad y aceleración las dos primeras siendo representadas en coordenadas polares.

Conclusiones

Las distribuciones que satisfacen de mejor manera las variables modeladas fueron la distribución Burr de tipo III, para la magnitud de velocidad y aceleración, y la distribución normal para el ángulo, de esta forma se obtienen las mediciones diarias; con un modelo de señal no estadístico ajustado a la amplitud, dada por el modelo de la aceleración, mas una gaussiana, para simular ruido, se obtienen las mediciones continuas.

Conclusiones

Se implementó una API Web, para facilitar la accesibilidad al modelo, con el Frameworks FastAPI y dos endpoints que entregan la información especificada en el Url y en los parámetros de configuración, nivel de daño (un número del 0 al 10) y el identificador único del motor que tiene asociado.

Conclusiones

El análisis en frecuencia se realiza mediante una transformada rápida de Fourier (FFT) lo que permite llevar la señal discretizada (mediciones a tiempo real) al dominio de la frecuencia y posteriormente graficarla; todo esto se logra mediante las librerías **numpy** y **matplotlib** de Python.

Conclusiones

Se utilizó una base de datos no relacional, **MongoDB**, en forma de microservicio con la empresa **MongoDB Atlas**; esta base de datos llamada **tesis** contiene 2 colecciones, una se encarga de almacenar la información de las mediciones diarias y la otra de almacenar los identificadores únicos que representan a los motores de los que se posee información.

Conclusiones

La automatización de procesos es fundamental para agilizar las tareas y la forma mas sencilla y eficiente de hacerlo fue mediante un Script.

Conclusiones

Se crearon 2 microservicios:

- ▶ servidor de sensorica
- ▶ servidor Web,

Conclusiones

Facilitar la utilización del modelo estadístico y la conexión con el microservicio de **sensorica** se implementó con un **cliente de sensorica** escrito en Go el cual permitió especificar las características inherentes al motor (Identificador del motor y de los sensores, características, etc) y al daño existente en el mismo (en una escala del 1 al 10), además, especificar la dirección IP, o puerto local, al cual se conectará.

Conclusiones

Se crearon 3 vistas para el cliente Web, estas son:

- ▶ **General**, con el cual se conoce el estado de todos los motores registrados en la base de datos, mediante paginación en una ventana se muestran 12 motores por vez, y controlar cuáles son mostrados, y especificar el identificador único del motor del cual se quiere obtener mas información.
- ▶ **Específica**, muestra la evolución histórica del motor mediante gráficas y una tabla exportable a Excel; adicionalmente permite solicitar la vista exhaustiva.
- ▶ **Exhaustiva**, solicita mediciones a tiempo real de la aceleración del motor y las descompone en frecuencia, permitiendo observar una gráfica de las frecuencias y magnitudes en las que vibra el motor; adicionalmente muestra toda la información de la vista específica.

Conclusiones

Se probó manualmente el sistema en los puntos claves e interconexiones, con lo cual se comprobó:

- ▶ La unicidad de la información.
- ▶ Fallas de conexión con el microservicio de base de datos.
- ▶ Solicitud de información a tiempo real a un motor sin conexión establecida.
- ▶ Peticiones invalidadas en el servidor Web

Conclusiones

Se comprobó el cumplimiento de los requerimientos mínimos que satisfacen las condiciones planteadas, mediante la demostración de la clasificación en los distintos niveles de daño, se corroboró la coherencia y legibilidad de las gráficas históricas y de la gráfica del análisis en frecuencia.

Recomendaciones

Utilizar un sistema de adquisición de datos, sensores o placas diseñadas para la obtención de las mediciones diarias, por ejemplo los datos a tiempo real para alimentar el sistema.

Recomendaciones

Implementar un sistema de autenticación y seguridad para que solo los usuarios con los privilegios correctos puedan acceder a la información, esto probablemente requiera el uso de otra base de datos preferiblemente relacional, y de esta forma poder hacer uso empresarial de la aplicación.

Recomendaciones

Un sistema de notificaciones configurable por el usuario, puede ser una gran ayuda al operador de la aplicación en el diagnóstico de fallas de forma temprana, para observar si algún motor recibe una medida fuera de los valores normales.

Recomendaciones

Debido a la complejidad de las conexiones HTTP2 una alternativa es el uso de Websockets los cuales permiten una conexión bidireccional sin la complejidad del nuevo protocolo evaluando los requerimientos de la aplicación. Otra alternativa, en un futuro cercano, posiblemente lo sea el HTTP3 la cual está siendo discutida por el comité del estándar HTTP para corregir fallas de la última versión.

Recomendaciones

Creación y uso de pruebas automatizadas para minimizar el tiempo de depuración y evitar fallas en producción, cabe destacar que esto implica un coste en el tiempo de desarrollo dedicado al diseño de las pruebas. La elección de usarlas dependerá de la experiencia previa del programador con este tipo de metodología de desarrollo.