

FREIREMAR

Gestión de Lances

Versión 0.1

05 de Abril de 2023



**Índice**

[1. Introducción 4](#_Toc132015396)

[2. Ventajas 5](#_Toc132015397)

[Anticipación 5](#_Toc132015398)

[Reducción de burocracia 5](#_Toc132015399)

[Control 5](#_Toc132015400)

[Visibilidad 5](#_Toc132015401)

[Trazabilidad 6](#_Toc132015402)

[Configuración remota 6](#_Toc132015403)

[Extensibilidad 6](#_Toc132015404)

[Integración 6](#_Toc132015405)

[3. Detalles técnicos 7](#_Toc132015406)

[3.1. Funcionalidades 8](#_Toc132015407)

[4. Aplicación de Lances 9](#_Toc132015408)

[4.1. Login 9](#_Toc132015409)

[4.2. Menu 9](#_Toc132015410)

[4.3. Dashboard 9](#_Toc132015411)

[4.4. Buque 9](#_Toc132015412)

[4.5. Artículos 9](#_Toc132015413)

[4.6. Clasificaciones 9](#_Toc132015414)

[4.7. Mareas 9](#_Toc132015415)

[4.8. Crear nuevo lance 9](#_Toc132015416)

[4.9. Resumen de lances de hoy 9](#_Toc132015417)

[4.10. Lances 9](#_Toc132015418)

[4.11. Resumen de Nota Pesca Diaria 9](#_Toc132015419)

[4.12. Nota Pesca Final 9](#_Toc132015420)

[5. Aplicación de Central 10](#_Toc132015421)

[5.1. Login 10](#_Toc132015422)

[5.2. Menu 10](#_Toc132015423)

[5.3. Dashboard 10](#_Toc132015424)

[5.4. Buques 10](#_Toc132015425)

[5.5. Artículos 10](#_Toc132015426)

[5.6. Clasificaciones 10](#_Toc132015427)

[5.7. Mareas 10](#_Toc132015428)

[5.8. Resumen de lances de hoy 10](#_Toc132015429)

[5.9. Resumen de Nota Pesca Diaria 10](#_Toc132015430)

[5.10. Nota Pesca Final 10](#_Toc132015431)

[6. Backend 11](#_Toc132015432)

# **Introducción**

Esta propuesta consiste varias aplicaciones que optimizarían la gestión de las notas de pesca diarias de los buques. Pero el trasfondo de esta aplicación es agilizar la mayor parte de los procesos burocráticos y la documentación de cada marea, además de tener una respuesta más rápida y más control sobre las capturas diariamente.

Se pretende que en los buques se realice una gestión de las capturas de una manera uniforme atendiendo a las particularidades de cada buque; debido a su tipología y modalidad de pesca, país donde opera.

El objetivo de este desarrollo no solo consiste en estandarizar la gestión sino también en acelerar y simplificar muchos procesos ineficientes, repetitivos o manuales que se hacen actualmente. Para ello estamos estudiando algunos casos reales en buques concretos y después analizaremos cómo podríamos aplicarlos a otros buques.

La aplicación aporta estas funcionalidades básicas:

* Generación automática de la nota de pesca diaria.

Al enviarse los lances se elimina la necesidad de enviar notas de pesca diarias ya que se generan automáticamente las notas de pesca diaria a medida que vayan guardando datos sin tener que manejar el correo.

* Generación automática de la nota de pesca final.

# **Ventajas**

Anticipación

Tener la información actualizada diariamente sin intervención del patrón de costas en la central podría hacer que pudiéramos vender mercancía desde el momento de la captura, anticipándonos así a la llegada a puerto con su consiguiente descarga, dándonos así una ventaja competitiva frente a posibles competidores, tratando el buque como un almacén más.

Podrían estimarse precios de artículos para maximizar el rendimiento de la marea, adaptando la estrategia de la pesca al status quo del mercado en ese preciso momento.

Reducción de burocracia

Actualmente la legislación obliga a la presentación electrónica de las capturas en el DEA (https://www.mapa.gob.es/es/pesca/temas/vigilancia-pesquera/manual-cumplimentacion-deav4\_tcm30-502219.pdf) el cual es un proceso repetitivo que podría calcularse automáticamente desde las notas de pesca diarias.

Certificados sanitarios semi automáticos.

Generación automática de nota de pesca final.

Control

Durante las descargas se hace una nota de pesca final que podríamos utilizar para verificar que lo declarado se corresponde con lo descargado, añadiendo así una herramienta más de control. Igualmente, durante este proceso de descarga se hacen inspecciones sanitarias y declaraciones a las autoridades marítimas que comprueban que todo está en orden, con lo que reduciríamos la posibilidad de sanciones por descuadres entre lo declarado en el DEA y lo descargado.

Visibilidad

Tendríamos una visión actualizada de lo que sucede en el barco y una ventana que nos permitiría añadir funcionalidades extra a la declaración de notas de pesca. El propio buque podría ver si se cumplen las expectativas de la marea visualizando la fuerza de la marea y la valoración de la misma en un cuadro de mandos.

El rendimiento de la marea estaría centralizado en un solo sitio que se alimentaría de múltiples fuentes que afectan al mismo, lo que proporcionaría una herramienta para detectar a tiempo posibles cambios de estrategia.

Se podría construir un dashboard con:

* Fuerza de la marea
* Nivel de capacidad en bodegas ocupada
* Ocupación en peso por clasificación
* Ocupación en cajas por clasificación
* Días restantes vs estimación de vuelta a puerto en días
* Valoración de la marea según gastos y capturas
* Media de Kg/día
* Media de €/día

Trazabilidad

Se compararían mareas anteriores en términos de productividad y/o valor de la marea.

Podrían añadirse funcionalidades como geolocalización y seguimiento, envío de logs de errores de la aplicación.

Configuración remota

Tendríamos la posibilidad de configurar las mareas remotamente, por ejemplo: clasificaciones, artículos.

Extensibilidad

Podríamos ampliar la funcionalidad de la aplicación mediante terminales satélites que se conectarían a un servidor local. Dichos satélites podrían ser tablets o dispositivos móviles que podrían utilizarse en diferentes localizaciones del buque.

Integración

La aplicación podría crecer para integrarse con otros procesos en la central inherentes en la propia marea, por ejemplo: aprovisionamientos (pedidos), gestión de tripulantes (rrhh), entrepot, etc.

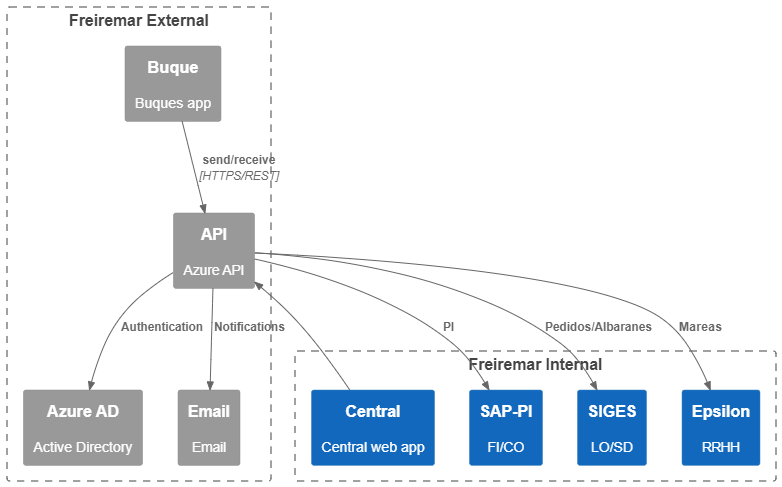
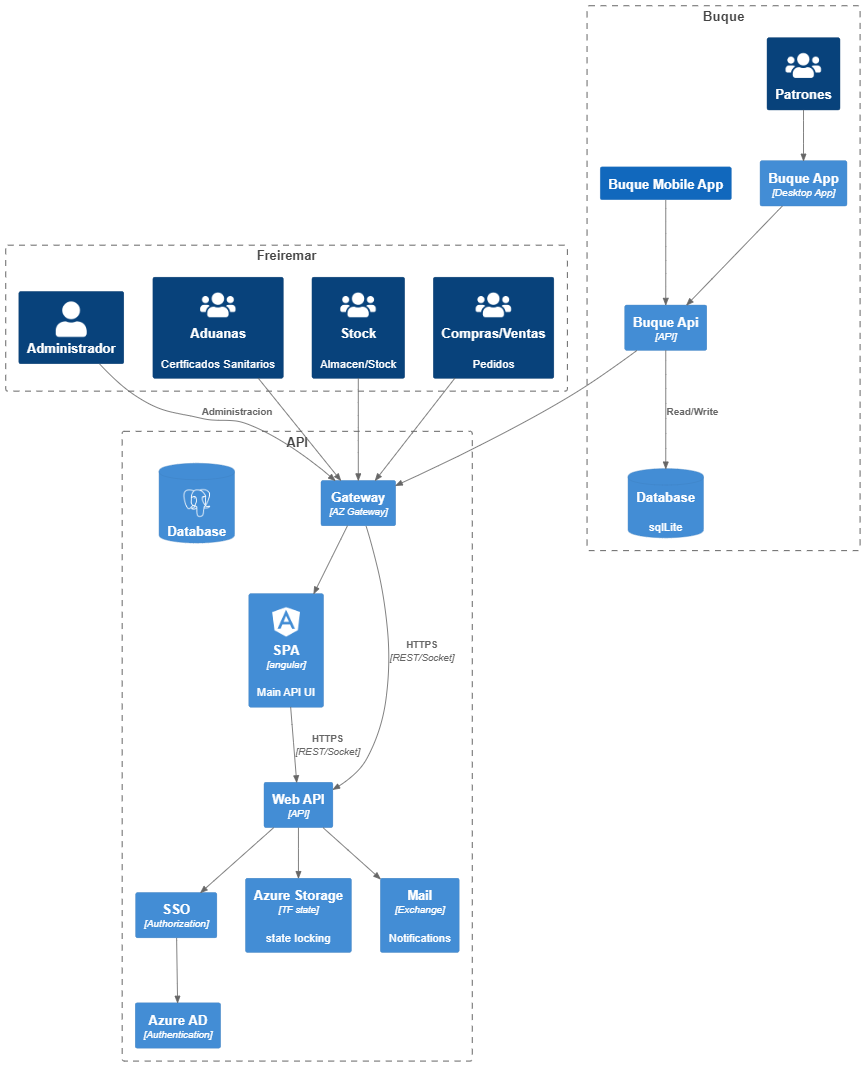
# Detalles técnicos

Según los usuarios consultados (Hector Cayon e Ivan Mckenzi), las comunicaciones de los buques tienen conexión permanente a internet por lo que hemos simplificado el protocolo de comunicación para usar REST vía HTTPS.

Dicha conexión, al ser satelital, padece de una latencia muy alta además de no estar garantizada, lo que hace que la solución deba ser una aplicación de escritorio en el buque. Otra ventaja de que sea una aplicación de escritorio es que reduciríamos el consumo de ancho de banda y evitaríamos mantener una conexión permanente para poder trabajar. Esto implica que la aplicación contará con una base de datos local (a determinar según decidamos en el backend) con datos básicos y maestros con los que puedan trabajar offline.

La aplicación deberá ser capaz de hacer login sin necesidad de estar conectada mediante un usuario local, pero si necesitará autenticarse debidamente al hacer cada envío de información. La recepción de información desde la central podemos diseñarla de manera no asistida mediante la instalación de un servicio en Windows o mediante los pipelines de Azure.

Cada buque tendrá su cuenta de correo y se comunicará directamente al servidor. Dicho buque tendrá un hash único para poder identificar unívocamente cada buque y así actualizar todas las aplicaciones que pueda estar ejecutando. Dicho Hash podría utilizarse con GPG para garantizar la autenticidad del remitente de cada petición, pudiendo así asegurar ambos endpoints de la comunicación.

Ciclo de vida de la aplicación:

1. Instalación vía AnyDesk o presencialmente en cada buque.
2. A partir de los datos de clasificaciones de la base de datos de Mareas de la aplicación de escritorio de la Central, construiremos los datos maestros básicos de la aplicación; Países, Puertos, Monedas, Tipos de Buque, Presentaciones, Envases, etc.
3. Igualmente, cada buque tendrá su propio maestro local de Orden-Familia-Especie-Articulo-Clasificación.
4. Cada buque tiene un correo electrónico para las comunicaciones, dicho correo está encriptado en un hash SHA256 que será su identificador para todos los comandos que reciba, el cual viajará en el Header de cada petición REST.
5. Se inicia la aplicación y si detecta que la base de datos no existe aún genera los datos más esenciales para iniciarse; entre ellos el login por defecto.
6. Una vez logado correctamente si detecta que la base de datos no está completa, la regenera a partir de los datos por defecto.
7. La arquitectura de la aplicación seguirá el patrón CQRS:
   1. Obtención de datos (obtención de logs, lances, etc.)
   2. Modificación de datos (inserción, borrado o modificación)
8. En Azure correrá un servicio REST encargado de recibir todos los datos enviados por los buques y almacenarlos en una base de datos a determinar según precio en Azure.
9. La autenticación se hará mediante un servicio OAUTH que se integrará con el actual Active Directory de Azure.

## Funcionalidades

El servicio contará con la siguiente biblioteca de comandos:

* Login: Autenticacion local y remota.
* Marea: Gestión de datos de la marea.
* Buque: Gestión de datos del buque.
* Lance: Gestión de lances.
* NPF: Generación/Envío de Nota pesca final.
* RESET: Reinicia la aplicación al estado de fábrica borrando todos los datos.
* SENDLogs: Envía a la central logs de la aplicación en caso de error.
* Conf: Envío remoto de la configuración de la app (login/pass, loglevel, etc.).

Cada envío deberá auditarse y tendrá estos estados de envíos:

* No Enviado
* Enviado, no recibido
* Enviado y recibido

# Aplicación de Lances

Aplicación stand-alone de escritorio a definir si queremos que sea multiplataforma o Windows capaz de trabajar offline y de comunicarse full dúplex con el api de la central.

## Login

## Menu

## Dashboard

## Buque

## Artículos

## Clasificaciones

* Envases
* Elaboraciones
* Presentaciones

## Mareas

## Crear nuevo lance

## Resumen de lances de hoy

## Lances

## Resumen de Nota Pesca Diaria

## Nota Pesca Final

# Aplicación de Central

Aplicación web servida desde Azure.

## Login

## Menu

## Dashboard

## Buques

## Artículos

## Clasificaciones

* Envases
* Elaboraciones
* Presentaciones

## Mareas

## Resumen de lances de hoy

## Resumen de Nota Pesca Diaria

## Nota Pesca Final

# Backend

El backend correrá en Azure y está en proceso de definición ya que estamos estudiando cómo sacar el mejor partido a la nube de Azure. Desconozco la nube de Azure por lo que necesitaré verlo más en profundidad, pero a priori podría anticipar que podemos utilizar:

* [Azure pipelines](https://azure.microsoft.com/en-us/products/devops/pipelines)
* Apps
* [CosmosDB](https://learn.microsoft.com/en-us/azure/cosmos-db/introduction)/Postgres
* [Azure Functions](https://learn.microsoft.com/en-us/azure/azure-functions/functions-overview)
* Notifications
* Azure Blob Storage