



Avance 2da Parte

Proyecto “EPA Móvil”

Autores: Patricio Chang Reyes
Francisco Pantoja González

Asignatura: Proyecto IV
Nombre Empresa: Empresa Portuaria Arica
Profesor: Diego Aracena Pizarro

Contenidos

- Correcciones fase anterior
- Tecnologías a Utilizar
- Alcance de Producto con Tecnologías
- Diagramas
- Modelamiento de Capa de Datos
- Arquitectura de Software
- Avances

Objetivos General y Específicos

Objetivo General

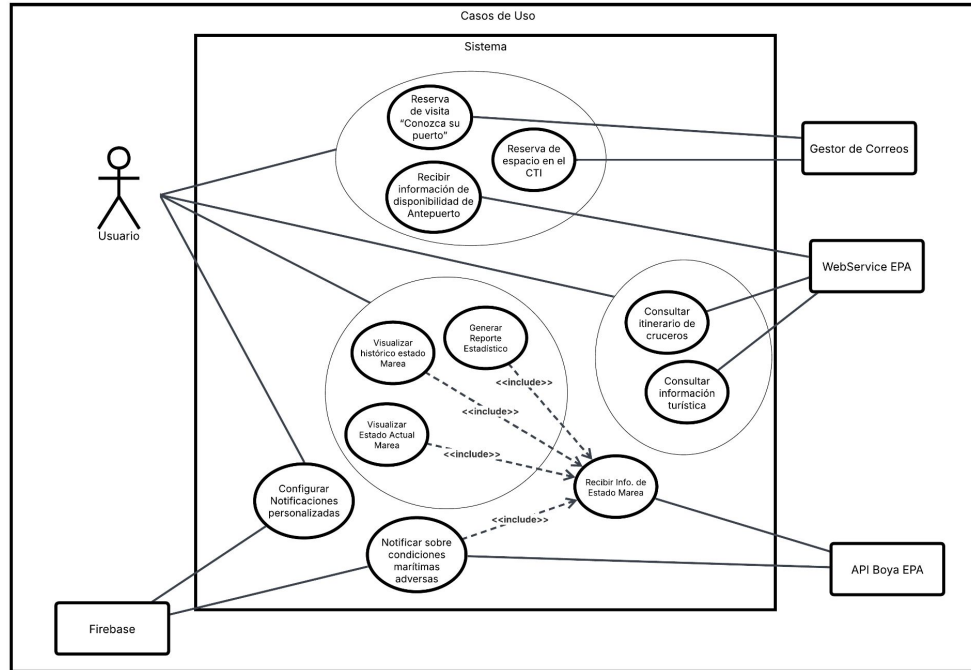
Dar solución a una problemática real desarrollando la aplicación móvil, "EPA Móvil" , que integre información técnica, operativa y turística para mejorar el acceso a los servicios de la Empresa Portuaria Arica.

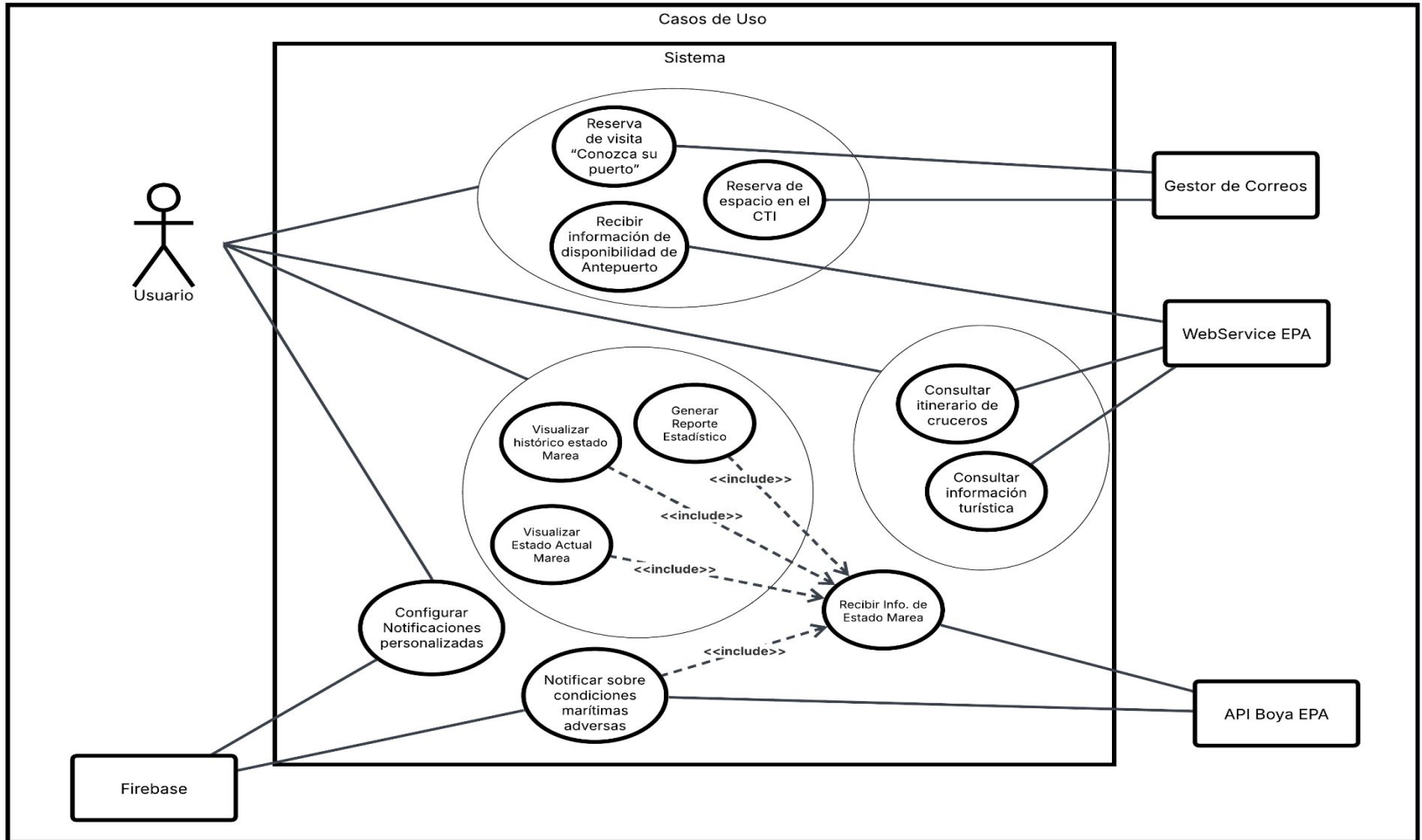
Objetivos General y Específicos

Objetivos Específicos

- Especificar el planteamiento, análisis, diseño y planificación del desarrollo de la aplicación “EPA Móvil”, definiendo el contexto, problemas, requisitos y etapas necesarias para su implementación.
- Implementar cada uno de los módulos necesarios para garantizar el correcto funcionamiento del sistema en base a sus requisitos iniciales, alcances y restricciones.
- Probar y testear las distintas iteraciones de producto de software en colaboración con el cliente, recopilando observaciones y propuestas de mejora razonables.
- Consolidar la iteración final de software para el periodo de trabajo determinado por la asignatura, poniendo el producto a disposición de la empresa y cliente.

Diagrama de CUS corregido





Frameworks a Utilizar



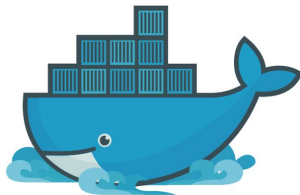
Compose
Multiplatform

Develop stunning shared UIs
for Android, iOS, desktop, and web.

JETBRAINS

 Kotlin

 nest



Componente	Framework	Lenguaje Base	Razón
Aplicación Móvil	Kotlin Multiplatform Mobile (CMP/KMP)	Kotlin	Permite un único desarrollo de la lógica de la aplicación con el framework Kotlin Multiplatform, luego la interfaz de usuario utiliza CMP, basado en Jetpack Compose. El nuevo estándar de codificación de interfaces gráficas de manera declarativa
Backend Ligero (Proxy)	NestJS	Typescript	Al estar lleno de funcionalidades permite crear soluciones flexibles y a medida según los requerimientos del usuario. Además, permite la integración de otro tipo de librerías relevantes para el proyecto como el SDK de Firebase y Redis.
	Docker	—	Permitirá el despliegue rápido y compatible en todas las plataformas donde se aloje el Backend del sistema.

Herramientas de Desarrollo



Firebase



Herramienta	Componente	Razón
Android Studio	Aplicación Móvil	Creado por Google en colaboración con JetBrains. Es el mejor entorno de desarrollo integrado para trabajar con aplicaciones móviles. Permite acceso a emuladores, visualizador de vista previa y depurador USB y Wifi para hacer pruebas
Visual Studio Code	Backend Ligero	Es uno de los entornos más completos para trabajar con lenguajes como Typescript debido a su alta cantidad de plugins e integraciones directas a la herramienta.
Firebase	Aplicación Móvil y Backend Ligero	Permite integrar la lógica de manejo de notificaciones y llamadas especiales del servidor backend a cada usuario de la aplicación móvil, evitando pasos extra en el desarrollo y garantizando que el envío de mensajes sea seguro para cada uno de los usuarios.
Uizard	Aplicación Móvil	Es una herramienta que permite realizar Mock-ups interactivos, los cuales permitirán validar las ideas del equipo de desarrollo en colaboración al cliente.
Lucid Charts	Aplicación Móvil y Backend Ligero	Permite el desarrollo de diagramas de manera rápida y ordenada.
Github	Aplicación Móvil y Backend Ligero	Permite guardar los estados y versiones del proyecto durante todo su ciclo de vida.

Repositorio de Github (en privado por el momento)

https://github.com/CHOP4NCHO/EPA_Movil/tree/dev

The screenshot displays the GitHub interface for the repository **EPA_Movil** (Private). The main content area shows the file explorer with folders: **backend**, **composeApp**, **gradle**, and **iosApp**. The current branch is **dev**, which is 24 commits ahead of **master**. The sidebar on the right is open to the **Collaborators and teams** section, showing the repository is private and has 1 collaborator: **PatricioChang**.

Collaborators and teams

Private repository
Only those with access to this repository can view it

Direct access
1 entity has access to this repository. [1 collaborator](#).

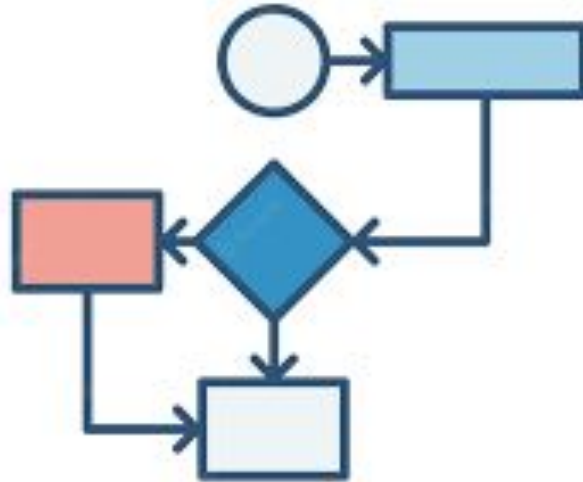
Manage access

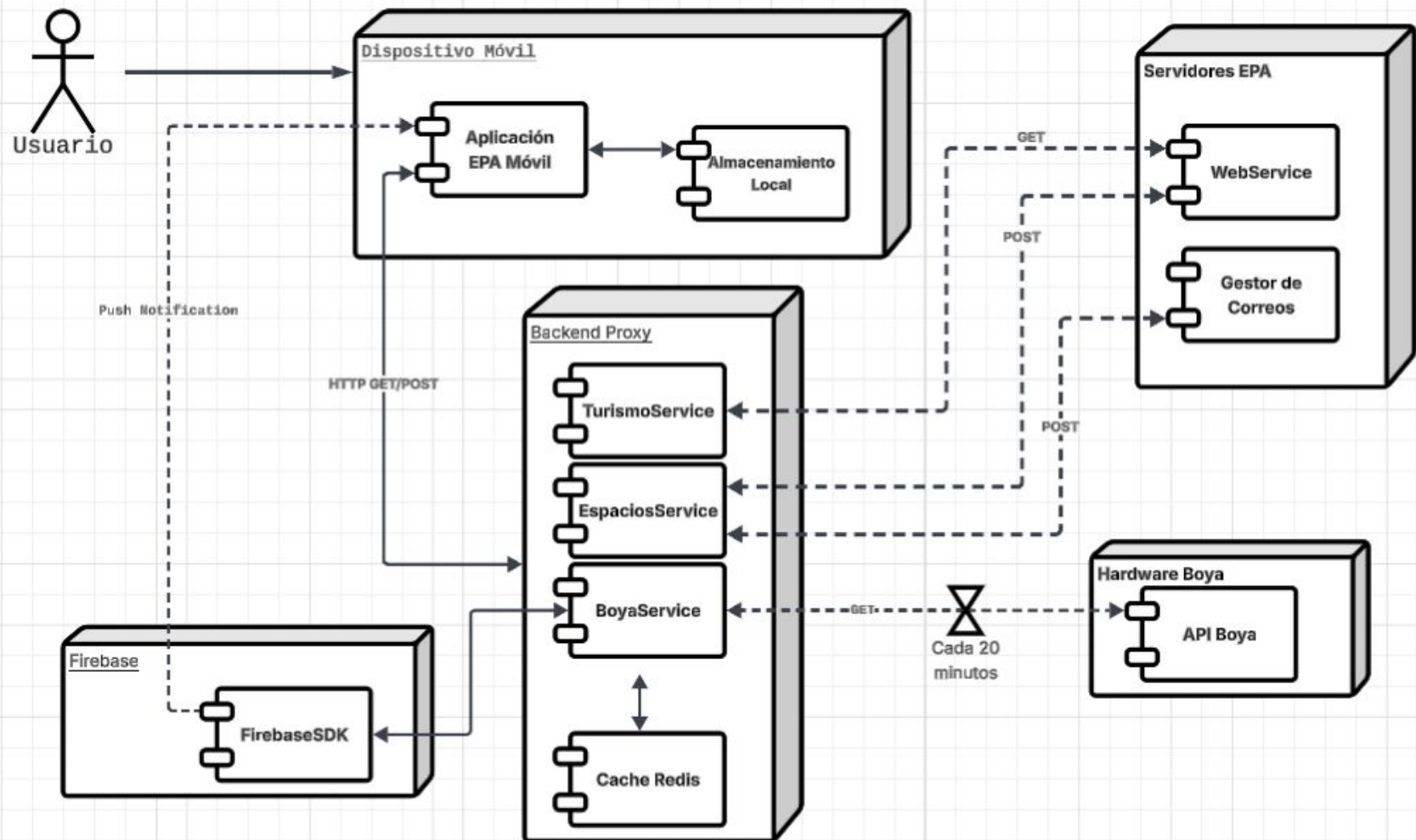
- Select all
- Find a collaborator...
- PatricioChang**
Collaborator

Alcances del Producto con Tecnologías Propuestas

ID	Tipo	Descripción
A1	Eficiencia	Las conexiones y comunicación entre cada uno de los componentes (app móvil - backend y viceversa) no deben exceder al mínimo de llamadas necesarias para garantizar la última información al usuario.
A2	Integración	Cada uno de los actores externos al sistema (Firebase, Gestor de Correos, Webservice EPA y la API Boya) deben ser integrados correctamente al componente de software que corresponda.
A3	Disponibilidad	La aplicación móvil debe funcionar adecuadamente tanto con como sin conexión a internet, haciendo uso del almacenamiento local. Por otro lado, el servidor debe operar correctamente incluso cuando no sea posible obtener la última información de la boya.
A4	Integridad	Cada uno de los componentes implementados de la aplicación deben coincidir con los diseños y modelamientos previamente hechos en cada una de las fases previas al desarrollo.
A5	Seguridad	El Backend debe actuar como un escudo de seguridad, centralizando y ocultando las credenciales sensibles (tokens de API Boya, credenciales SMTP) para que nunca queden expuestas en el código de la aplicación móvil.
A6	Portabilidad	El sistema Backend debe estar contenerizado mediante Docker para garantizar que pueda ser desplegado rápidamente y sin conflictos de dependencias en cualquier servidor que la Empresa Portuaria Arica disponga.

Diagramas de Componentes e Interacciones

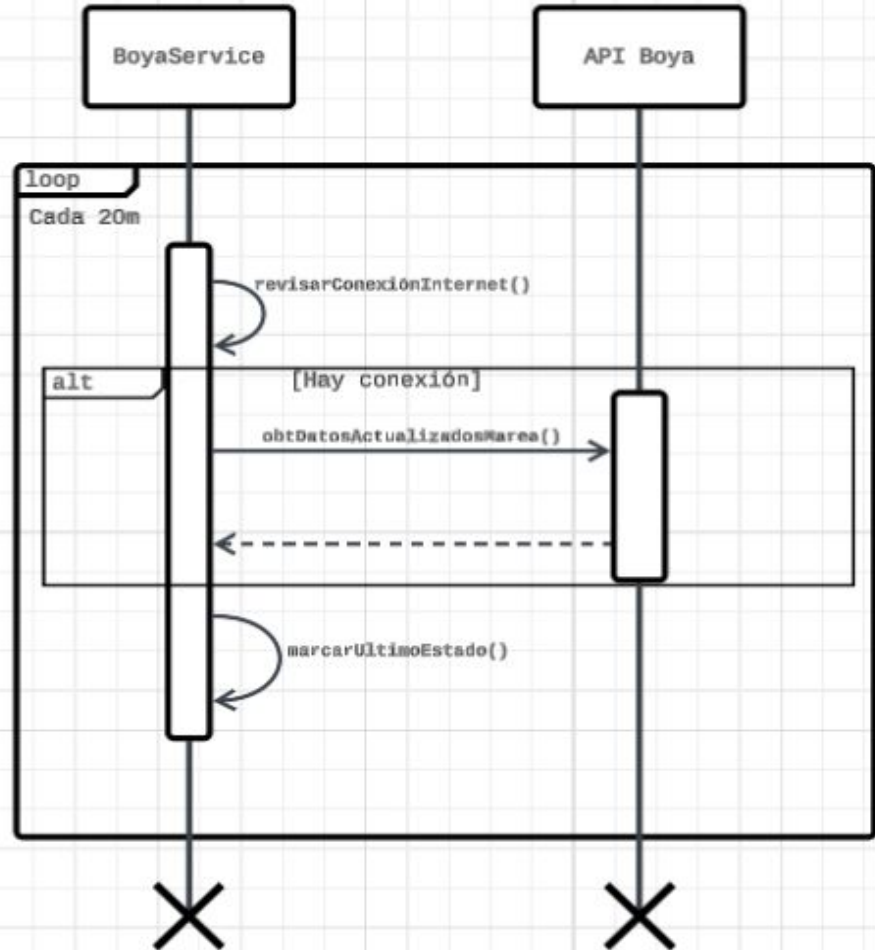




Subsistema de Boya

Uso de 2 diagramas de secuencia.

Se ha dejado fuera de la secuencia a Redis, ya que su intervención en el subsistema es más auxiliar que crítica.

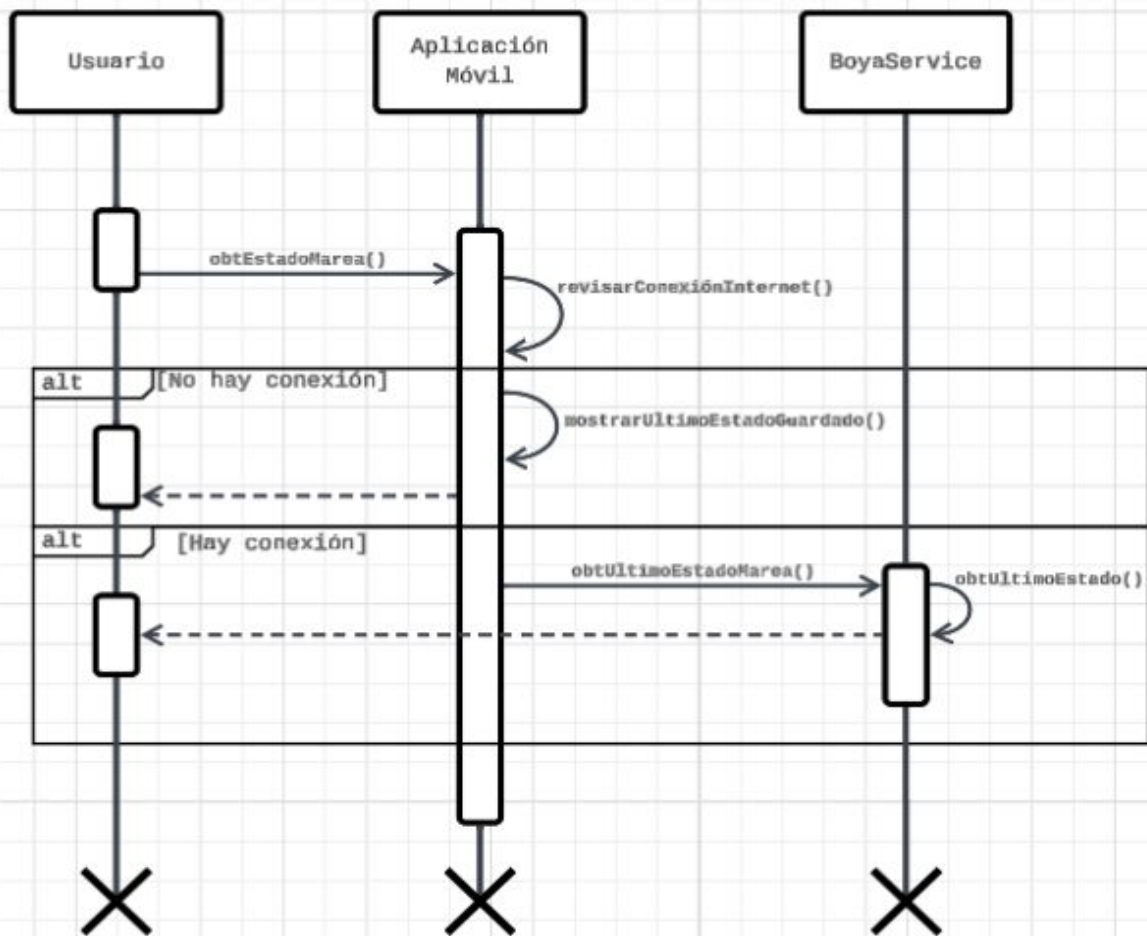


Subsistema de Boya

Uso de 2 diagramas de secuencia.

Se ha dejado fuera de la secuencia a Redis, ya que su intervención en el subsistema es más auxiliar que crítica.

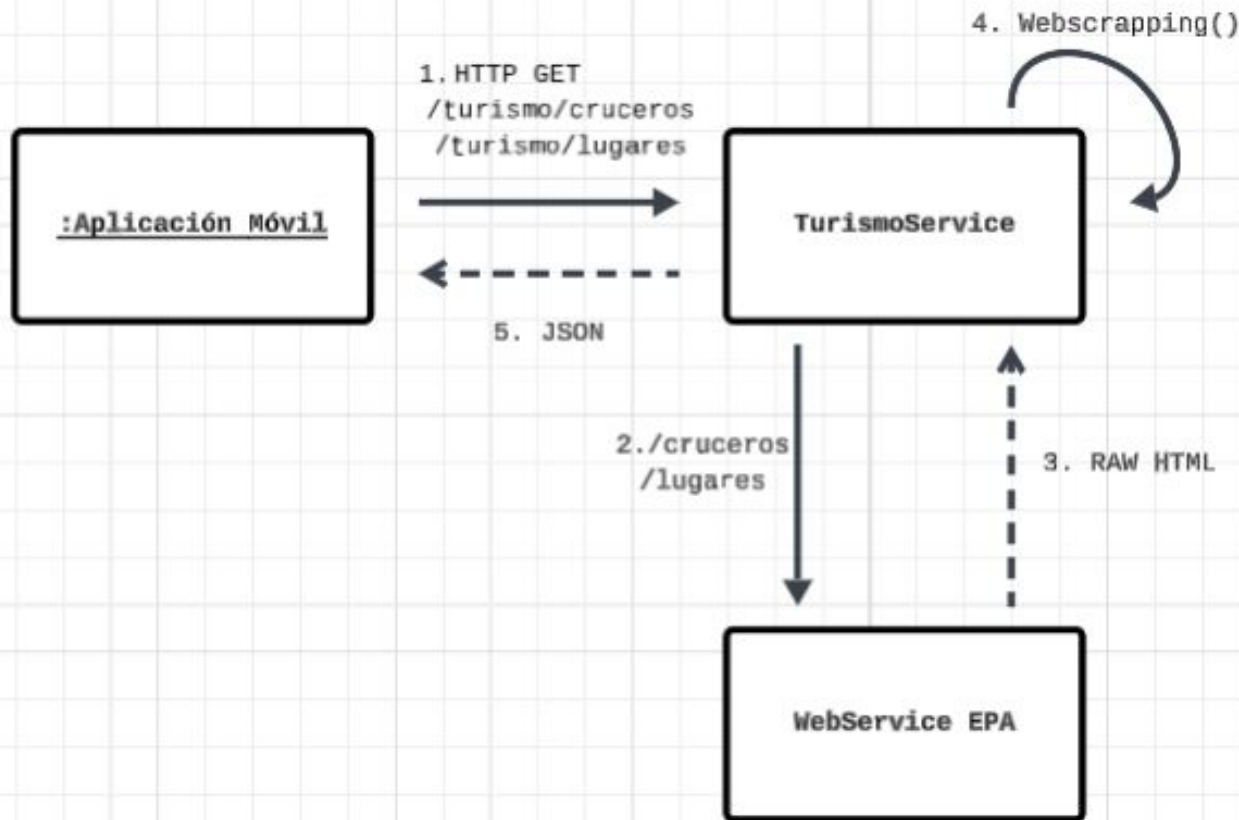
DIAGRAMA DE SEQ 2: Obtener Último Estado de la Boya (usuario)



Subsistema de Turismo

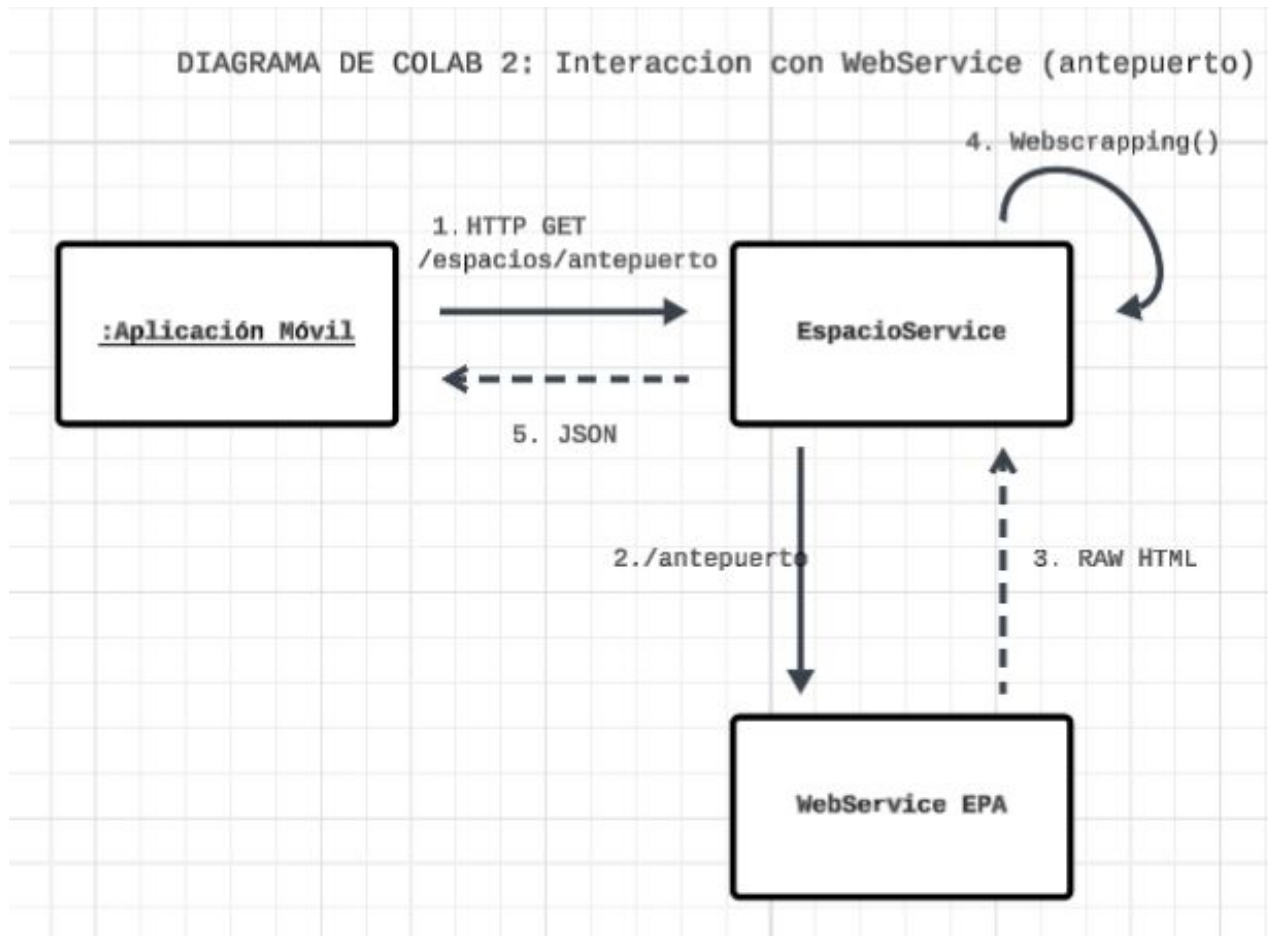
Uso de un diagrama de Interacción para ambos casos (turismo y cruceros)

DIAGRAMA DE COLAB 1: Interacción con Webservice (turismo)



Subsistema de Espacios

Uso de 2 diagramas, uno de secuencia y otro de interacciones. Para el estado de antepuerto y los formularios respectivamente.



Subsistema de Espacios

Uso de 2 diagramas, uno de secuencia y otro de interacciones. Para el estado de antepuerto y los formularios respectivamente.

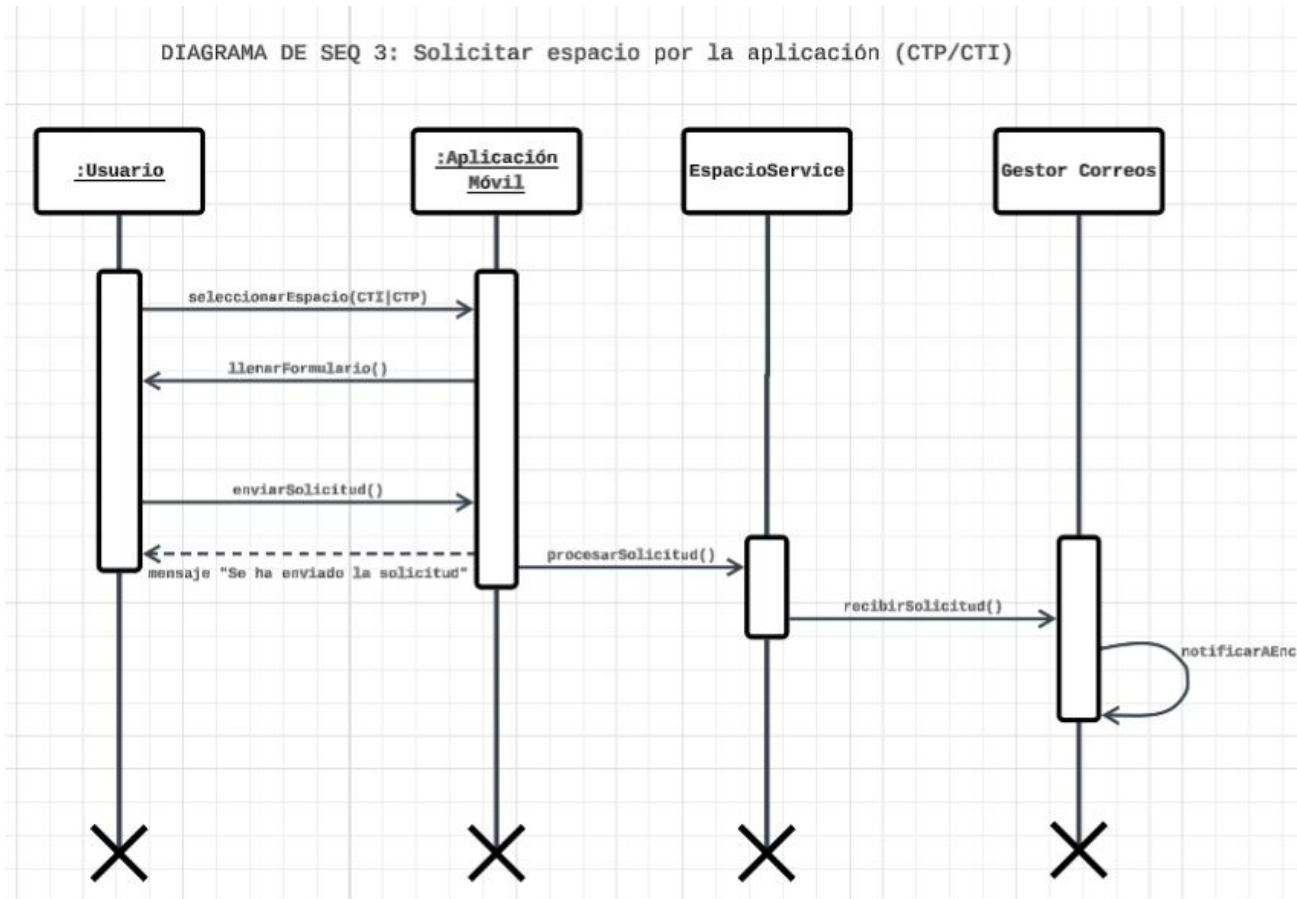
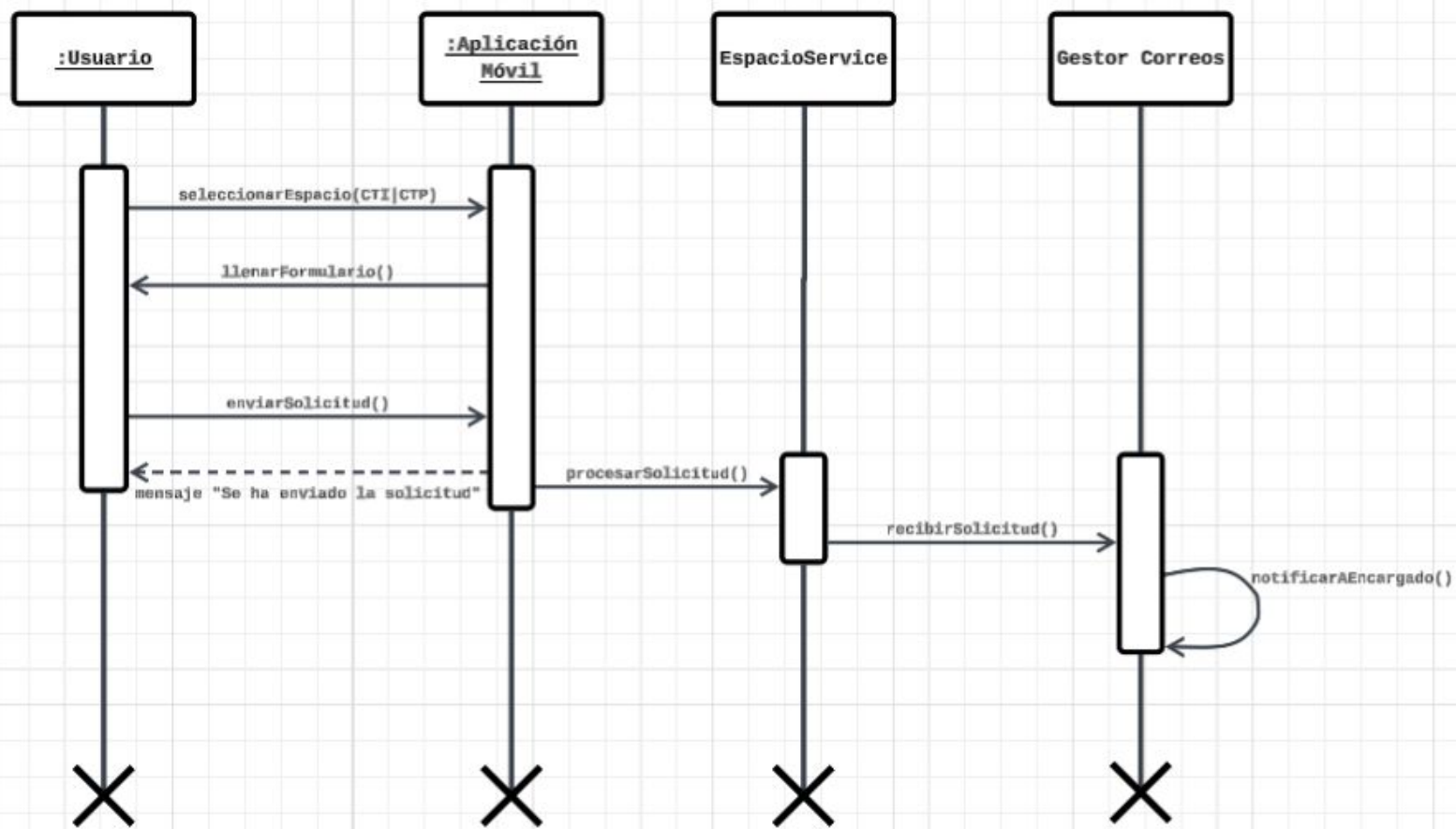
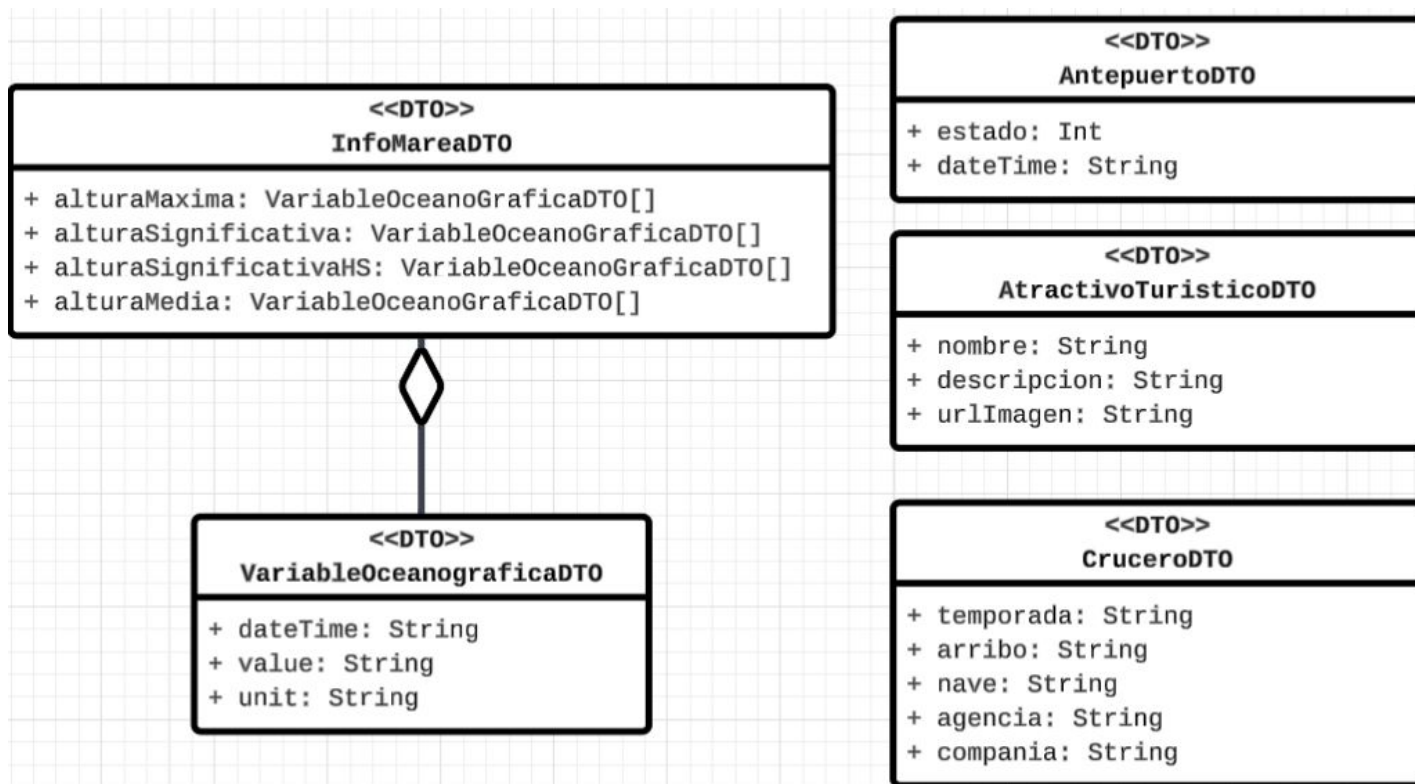


DIAGRAMA DE SEQ 3: Solicitar espacio por la aplicación (CTP/CTI)

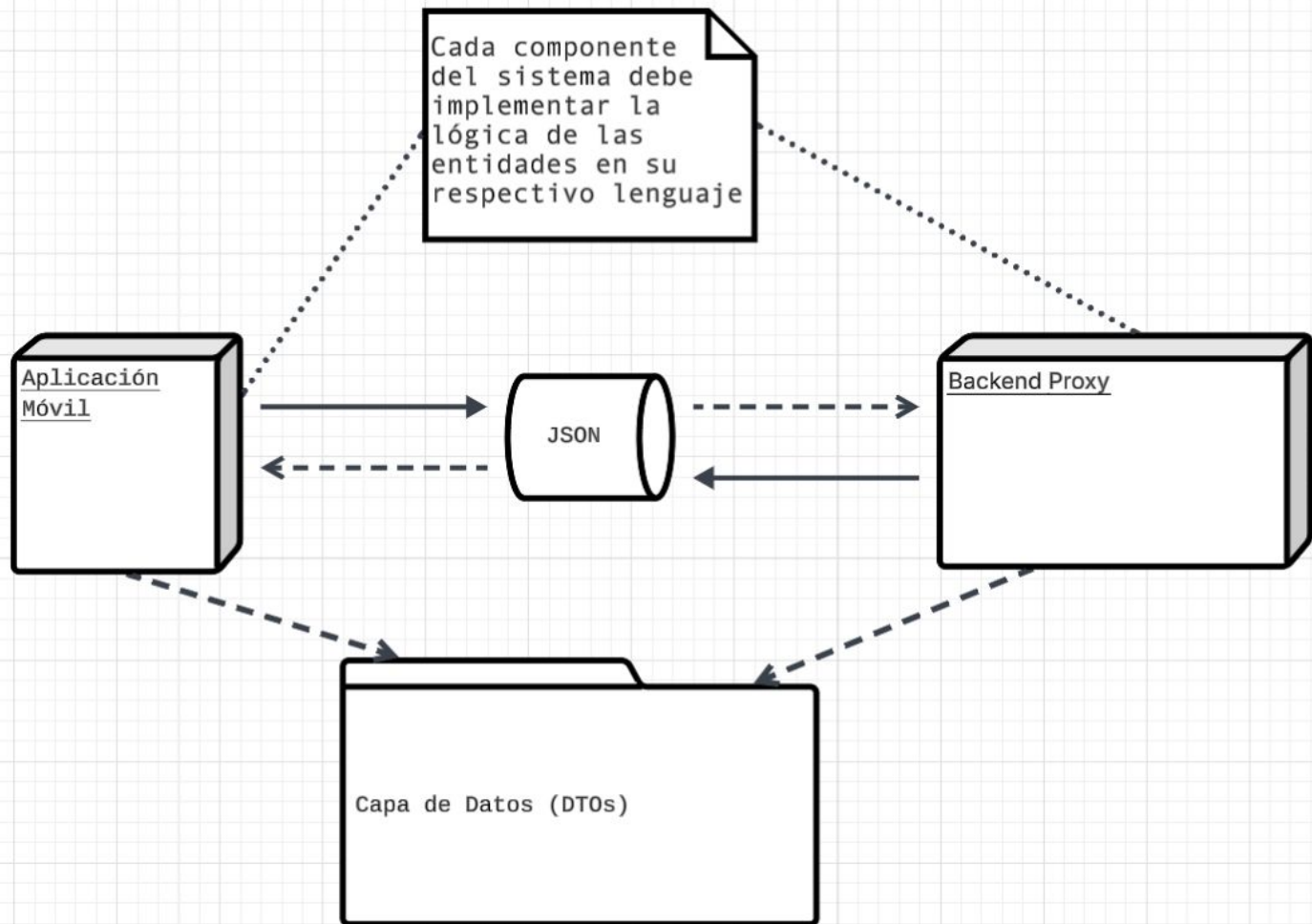


Modelo de la Capa de Datos



Modelo de la Capa de Datos

Comunicación haciendo uso de los DTO



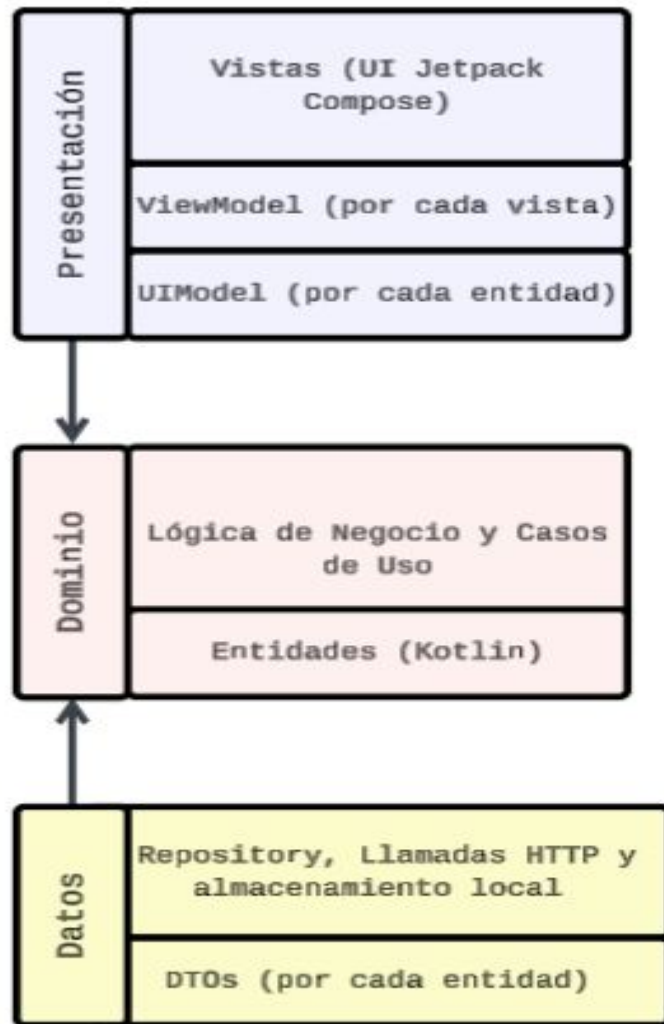
Modelo de la Capa de Datos

Información en crudo proveniente de la API

	Nombre variable	Definición	Unidad	Umbrales	Valores nulos
SENSOR DE OLEAJE	VMXL	Altura máxima	cm	0 a 3000	65535
	VH110	Altura significativa 1/10	cm	0 a 3000	65535
	VAVH	Altura significativa (Hs) 1/3	cm	0 a 3000	65535
	VHMO	Altura media	cm	0 a 3000	65535
	VTMX	Periodo máximo	seg	1 a 100.0	6553.5
	VTPK	Periodo ola pico	seg	1 a 100.0	6553.5
	VT110	Periodo ola significativa 1/10	seg	1 a 100.0	6553.5
	VAVT	Periodo ola significativa (Hs) 1/3	seg	1 a 100.0	6553.5
	VGTA	Periodo medio	seg	1 a 100.0	6553.5
	VDIR	Dirección del oleaje	º	0 a 359.9	6553.5
	VPSP	Dispersión del oleaje	º	0 a 359.9	6553.5
	VQTY	Cantidad de olas	Cantidad	1 a 1000	65535

Arquitectura de Software

(Aplicación Móvil)



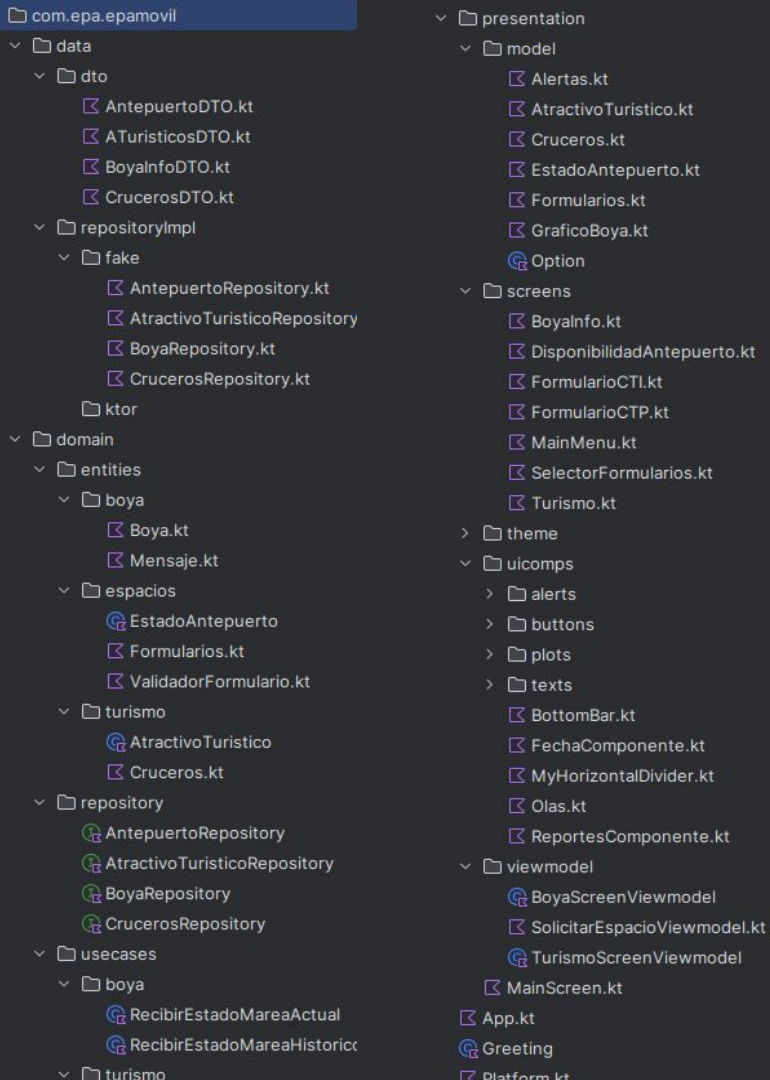
Avances (App móvil)

@Serializable

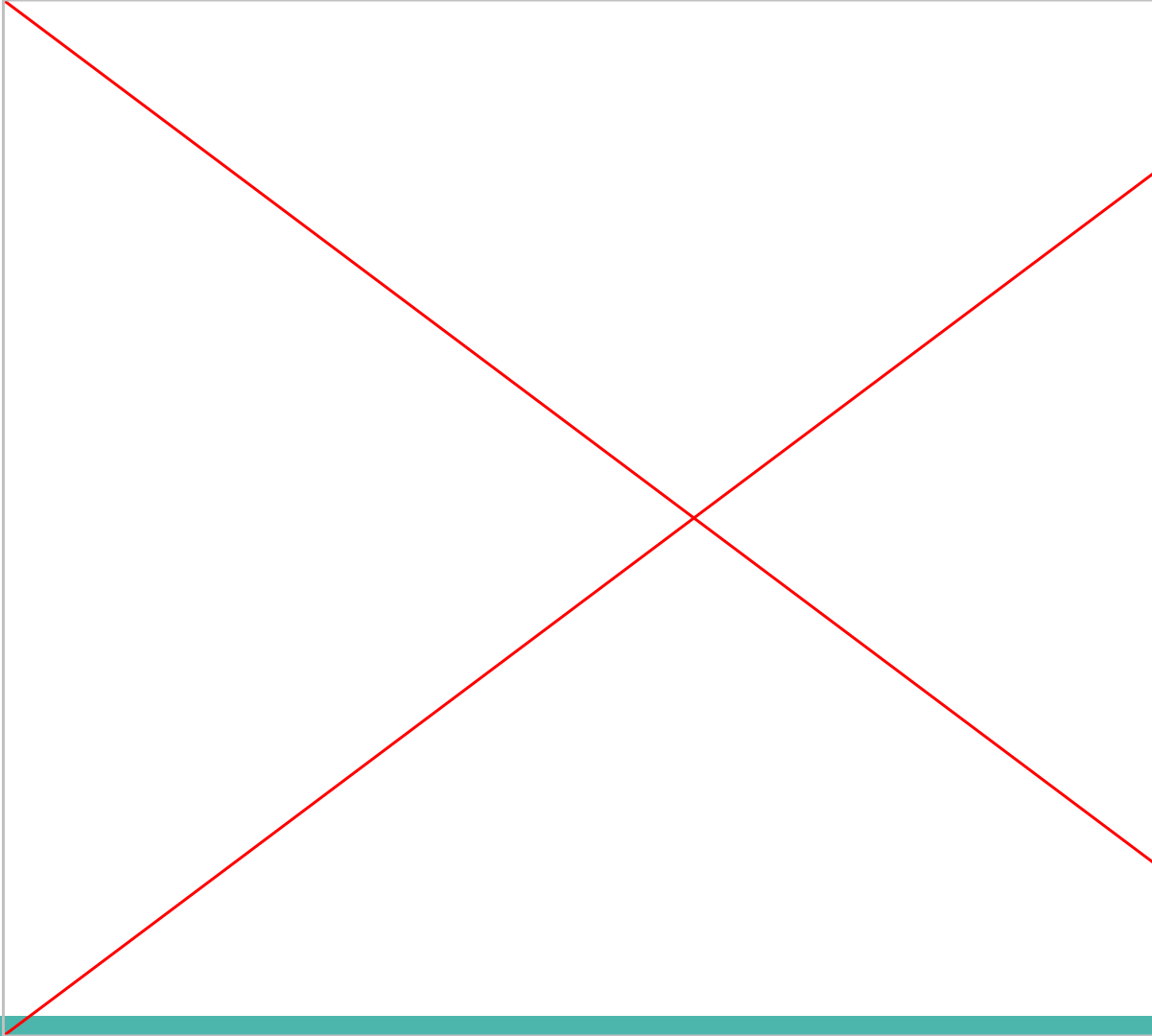
```
data class AtractivoTuristicoDTO (  
    val nombre: String,  
    val descripcion: String,  
    val urlImagen: String  
)
```

3 Usages

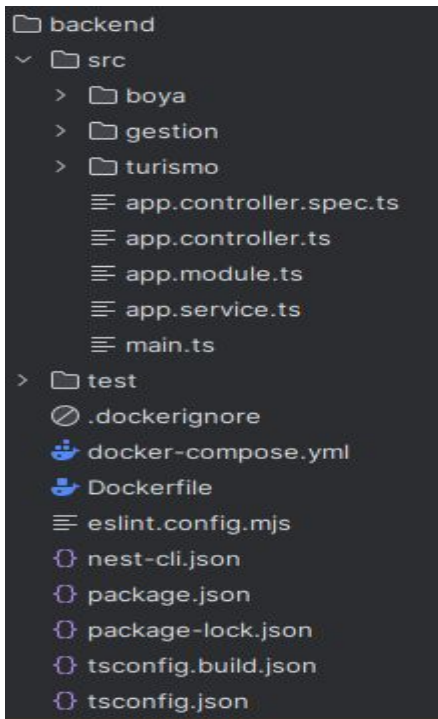
```
fun AtractivoTuristicoDTO.toAtractivoTuristico(): AtractivoTuristico {  
    return AtractivoTuristico(  
        nombre = nombre,  
        descripcion = descripcion,  
        imgReferencia = urlImagen  
    )  
}
```



Avances (App móvil)



Avances (Backend Ligero Proxy)



```
const cacheFilePath = path.join(process.cwd(), 'data', 'latest-data.json');const historyFile
const API_TOKEN = process.env.API_TOKEN;
const API_URL_WAVES = `https://oceancom.msm-data.com/api/device/10/Waves/Wave%20Height/?tok
const API_URL_WIND = `https://oceancom.msm-data.com/api/device/10/EMA/Wind%20Speed/?token=$
const API_URL_TEMP = `https://oceancom.msm-data.com/api/device/10/EMA/Temperature/?token=${

@Injectable()
export class BoyaService {
  private readonly logger = new Logger(BoyaService.name);

  constructor(private readonly httpService: HttpService) {
    this.logger.log('Iniciando servicio.. Actualizando datos por primera vez.');
```

```
    this.handleCron();
  }

  @Cron('*/*20 * * * *')
  async handleCron() {
    this.logger.log('EJECUTANDO CRON: Actualizando datos de la boya...');
```

```
    try {
      const [wavesResponse, windResponse, tempResponse] = await Promise.all([
        firstValueFrom(this.httpService.get(API_URL_WAVES)),
        firstValueFrom(this.httpService.get(API_URL_WIND)),
        firstValueFrom(this.httpService.get(API_URL_TEMP)),
      ]);

      // --- INICIO DE LA CORRECCIÓN DE LÓGICA ---
```

Avances (Backend Ligero Proxy)



The screenshot shows a web browser window with the address bar displaying `http://localhost:3000/boya/actual`. Below the address bar, there are three tabs: **JSON** (selected), **Datos en bruto**, and **Cabeceras**. Underneath the tabs, there are several action buttons: **Guardar**, **Copiar**, **Contraer todo**, **Expandir todo**, and **Filtrar JSON**. The main content area displays a JSON object with the following key-value pairs:

<code>altura0la:</code>	<code>1.13</code>
<code>velocidadViento:</code>	<code>3.23</code>
<code>temperatura:</code>	<code>16.9</code>
<code>ultimaActualizacion:</code>	<code>"2025-11-10T13:20:00.000Z"</code>

Avances (planificación ScrumBan)

Product Backlog ...

- [HU-05] Acceder a datos offline
- [HU-07] Ver mapa con puntos de interés
- [HU-11] Ver estado antepuerto (camioneros)
- [HU-14] Integrar Firebase para datos/notif.
- [HU-08] Información instalaciones del puerto

+ Añade una tarjeta

Sprint Backlog ...

- [HU-02] Alertas automáticas por malas condiciones
- [HU-06] Ver horarios de Cruceros
- [HU-10] Agendar visita "Conozca su Puerto"
- [HU-13] Interfaz intuitiva y fácil de usar

+ Añade una tarjeta

En Progreso ...

- [HU-01] Dashboard Boya (Ver variables clave)
- [HU-09] Reservar espacio CTI online

+ Añade una tarjeta

Testing

- [HU-03] Consultar histórico de datos oceanográficos
- [HU-04] Descargar reportes estadísticos (CSV)

+ Añade una tarjeta

Hecho ...

- [HU-12] Arquitectura multiplataforma (Android/iOS)

+ Añade una tarjeta

Conclusiones

1. Se validaron las decisiones arquitectónicas y de diseño claves.
2. Se implementó la funcionalidad básica de pruebas (backend) para demostrar la viabilidad de los requerimientos solicitados de la fase anterior.
3. Se aseguró la escalabilidad de la Aplicación Móvil siguiendo patrones y filosofías de desarrollo claras (Clean Architecture)
4. Se mitigó tempranamente el riesgo técnico más crítico: La conexión funcional a la API de la BOya por medio de Crontabs



Avance 2da Parte

Proyecto “EPA Móvil”

Autores: Patricio Chang Reyes
Francisco Pantoja González

Asignatura: Proyecto IV
Nombre Empresa: Empresa Portuaria Arica
Profesor: Diego Aracena Pizarro