**Documentación de la aplicación**

**"Web Agencia de Viajes García"**

**Formativo:** Máster en Programación Java & Máster en Full Stack Developer

**Módulo:** Máster de desarrollo

**Tarea:** TFM

**Alumno:** Pedro Donato García López

**Curso:** Junio 24.2

**URL (GitHub Repository**): <https://github.com/PedroDonato137/TFM.WebViajesGarcia>

**ATRIUM**

**Índice**

1. Título y descripción breve
2. Tecnologías utilizadas
3. Funcionalidades
4. Arquitectura de la aplicación
5. Manual de usuario
6. Configuración e implementación
7. Estructura del código
8. Base de datos
9. Futuras mejoras y mantenimiento
10. **Título y descripción breve**

**Título**: Web Agencia de Viajes García  
**Descripción**: Es una aplicación web diseñada para una agencia que ofrece cruceros por todo el mundo. Su objetivo principal es brindar información sobre viajes, noticias relacionadas y tipos de barcos, además de permitir a los usuarios registrarse y realizar pedidos de manera sencilla.

1. **Tecnologías utilizadas**

La aplicación ha sido desarrollada utilizando las siguientes tecnologías:

* Backend: Spring Boot, implementado en Java.
* Base de datos: MySQL, para la gestión y almacenamiento de datos.
* Frontend: Angular, con HTML y CSS para la interfaz de usuario.

1. **Funcionalidades**

La aplicación incluye las siguientes funcionalidades principales:

* + Mostrar información: Noticias relevantes sobre cruceros, tipos de barcos y descripciones de viajes.
  + Registro y autenticación: Los usuarios pueden registrarse, iniciar sesión y gestionar sus pedidos.
  + Realizar pedidos: Opción para seleccionar y confirmar reservas de cruceros.

1. **Arquitectura de la aplicación**

La aplicación utiliza múltiples arquitecturas de software:

* Arquitectura MVC: La estructura de datos está modularizada, separando la lógica de negocio (controladores), los modelos y las vistas.
* Arquitectura de microservicios: Actualmente se han desarrollado tres microservicios
  + Gestión de usuarios.
  + Gestión de pedidos.
  + Visualización de pedidos.
* Arquitectura cliente-servidor: El cliente realiza solicitudes al servidor, el cual interactúa con la base de datos para devolver los resultados solicitados.

1. **Manual de usuario**

Pasos para utilizar la aplicación:  
  
1. Registro: Los usuarios pueden registrarse proporcionando sus datos personales.  
2. Inicio de sesión: Una vez registrados, pueden iniciar sesión utilizando sus credenciales.  
3. Explorar: Navegar por las diferentes secciones para obtener información sobre cruceros, barcos y noticias.  
4. Realizar pedidos: Seleccionar un crucero y confirmar la reserva a través del sistema de pedidos.

1. **Configuración e implementación**

Es una aplicación web que no requiere instalación. La aplicación está desplegada en un servidor web accesible mediante un navegador.  
Requisitos del servidor:

* Java Runtime Environment (JRE) para ejecutar el backend.
* Servidor MySQL para la base de datos.
* Node.js y Angular CLI para mantener o modificar el frontend.

1. **Estructura del código**

El código está organizado según el patrón MVC (Modelo-Vista-Controlador)

* Modelo: Contiene la definición de las entidades y sus relaciones.
* Vista: Desarrollada en Angular, HTML y CSS para la interfaz de usuario.
* Controlador: Maneja las solicitudes HTTP y coordina la lógica de negocio.

1. **Base de datos**

La aplicación utiliza una base de datos MySQL para gestionar la información.

Esquema de la base de datos:

**Usuarios:**

id\_usuario(PK)

name

correo

password

rol

**Reservas:**

id\_reservas(PK)

idUsuario(FK)

fecha

status

**Viajes:**

id\_viaje(PK)

name

idContinente

capacidad

precio

**Continentes:**

idContinente (PK)

name

**ElementosReservas:**

id\_elementosReservas (PK)

id\_reservas(PK)

id\_viajes (PK)

unidades

1. **Futuras mejoras y mantenimiento**

- Nuevos microservicios: Se planea agregar microservicios para la gestión de barcos, reportes de actividad, la seccion de noticias dinamica con su tabla y noticias propias  
- Optimizaciones: Mejorar el rendimiento de las consultas a la base de datos.  
- Soporte: Los usuarios pueden reportar problemas o sugerir mejoras a través de un formulario de contacto en la aplicación.

- Mejorar control de información: poder dar de baja reservas y ver la información de perfil del usuario