

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Diseño de un sistema software multitenant para apoyar el negocio de mensajería express dirigido a pequeñas y medianas empresas.

Dilan Esteban Rey Sepúlveda y Silvia Alejandra Cárdenas Santos

Trabajo de Grado para Optar al Título de Ingeniero(a) de Sistemas

Director

Gabriel Rodrigo Pedraza Ferreira

Dr en Ingeniería de sistemas

Codirector

Jathinson Meneses Mendoza

MSc en Ingeniería de sistemas

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingeniería de Sistemas e Informática

Bucaramanga

2025

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Dedicatoria

A mi madre, Liliana Elena Sepulveda, quien ha sido mi mayor apoyo y compañía a lo largo de mi vida y es la fuente de mi fortaleza e inspiración para seguir adelante en todo momento. Has dedicado toda tu vida a mí con gran amor, ternura y templanza; te dedico este trabajo con el mismo amor gigante que me has compartido durante toda mi vida. Esto es una victoria de nosotros, gracias por tanto.

A mis abuelos, Blanca Guevara y Romulo Rey, quienes desde muy pequeño me han acogido con amor y complacencia, y quienes no dudaron ni un segundo en apoyarme cuando tomé la decisión de embarcarme en esta aventura académica fuera de mi ciudad natal. Sin su ayuda nada de esto hubiera sido posible y por toda mi vida les estaré eternamente agradecido.

A mi padre, Andres Rey, quien me dio la vida y siempre se preocupó por no permitirme que cometiera sus mismos errores. Gran parte de la persona que soy hoy en día es gracias a ti.

Los amo mucho a todos.

Dilan

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

A mi mamá, Silvia Cristina Santos, quien desde pequeña has hecho todo por y para mí. Me inculcaste lo que es el trabajo duro y gracias a ti aprendí lo que es la resiliencia. Has sido la fuente de mi fortaleza e inspiración para seguir adelante en todo momento. Quiero que sepas que este es el primer paso para un futuro brillante tal como esperas, y espero que podamos compartirlo juntas.

A mi tía, Adriana Rocío Santos, quien fue un gran apoyo en todos estos años y sin quien no sería posible estar donde estoy ahora. Eres uno de mis referentes más cercanos, ¡y mira dónde estamos ahora! Espero poder devolverte todo lo que hiciste por mí. Seamos felices en este futuro que se avecina.

A mi nona, Gilma Vesga, quien fue un pilar fundamental en todos estos años. Eres la referente de la familia, una persona tan especial a la que queremos mucho y que siempre está ahí en todos los momentos tanto buenos como malos.

Este logro es de nosotras. Esto es una victoria de todas. Las amo mucho.

Silvia

Agradecimientos

Quiero extender mis más profundos agradecimientos a nuestro director, el profesor Gabriel Rodrigo Pedraza Ferreira, quien no se lo pensó dos veces a la hora de presentarle esta idea y de decidir dirigirnos en la realización de este proyecto. He de decir que no me parece que hubiera habido un profesor más idóneo para esta tarea.

A Daya, quien tuve la oportunidad de conocer al iniciar esta trayectoria académica y quien me acompañó durante toda esta, especialmente en el último tramo y durante la realización de este proyecto. Te agradezco con mucho amor y cariño por tus palabras, consejos, apoyo y acompañamiento en las noches de vigilia y los momentos difíciles.

A mi compañera Silvia, quien puedo decir se convirtió en una de las pocas amistades que pude haber forjado en la universidad y estoy muy agradecido de haber podido llevar a cabo este proyecto junto a ella. Sin tu apoyo clave, este proyecto no habría llegado a realizarse de manera tan extraordinaria. ¡Está hecho!

A mis compañeros del trabajo, en especial a mi jefe Cris, quienes siempre me apoyaron y me permitieron la oportunidad de poder trabajar para subsistir y hacer este sueño realidad. Muchas gracias por su paciencia y comprensión.

Dilan

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

En primer lugar, quiero expresar mi más sincero agradecimiento al director de este proyecto, el profesor Gabriel Rodrigo Pedraza Ferreira, quien no dudó en aceptar la dirección de esta iniciativa y nos brindó su invaluable orientación.

En el ámbito personal, mi más profundo agradecimiento está dirigido a mi familia, quienes han sido mi refugio y fortaleza durante este largo camino. A Carlos Eduardo, William, Silvia Fernanda y Felipe, junto a otros familiares queridos que han creído en mí desde el primer día, sin sus palabras de aliento este sueño no habría sido posible.

Un reconocimiento especial merece Juan Pablo, quien se convirtió en mucho más que un acompañante en este proceso académico. Gracias por tus consejos certeros, por escuchar mis dudas con paciencia infinita y por ofrecerme esa retroalimentación constructiva que me permitió crecer como profesional y como persona.

Una mención especial para Dilan, quien más que un compañero de proyecto se convirtió en un verdadero amigo. Su colaboración fue fundamental para el desarrollo de este trabajo, pero sobre todo valoro la confianza que depositó en mí al incluirme en esta iniciativa. Gracias por las horas compartidas y por demostrarme que trabajar junto a un amigo hace que cada logro sea más significativo. ¡Lo logramos!

A todos los que han sido parte de esta historia: gracias por creer, acompañar e inspirar.

Silvia

Tabla de Contenido

	Pág.
Introducción	24
1. Planteamiento y justificación del problema	27
2. Objetivos	29
2.1 Objetivo General	29
2.2 Objetivos Específicos.....	29
3. Alcance	29
4. Marco de referencia	31
4.1 Marco conceptual.....	32
4.1.1 Arquitectura de software.....	32
4.1.2 Software como servicio (SaaS).....	32
4.1.3 Arquitectura multitenant	33
4.1.4 Mensajería express.....	37
4.1.5 Aplicaciones web SPA.....	38
4.2 Estado del arte.....	38
5. Metodología	41
5.1 Identificación de características.....	42
5.2 Definición de requerimientos.....	43
5.3 Diseño de la arquitectura	44
5.4 Implementación de la aplicación	45
5.5 Validación de calidad de software	45
5.6 Documentación	46

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

5.7	Sustentación de resultados	47
6.	Análisis y requerimientos del sistema.....	47
6.1	Enfoque seleccionado	48
6.1.1	Justificación técnica.....	48
6.1.2	Implementación específica.....	48
6.2	Alcance del sistema (instancia de la aplicación).....	49
6.3	Requerimientos del sistema (instancia de la aplicación)	50
6.3.1	Requerimientos funcionales.....	50
6.3.2	Requerimientos no funcionales.....	53
6.3.2	Requerimientos técnicos	54
6.4	Historias de usuario (HU)	55
6.4.1	Gestión de personal.....	55
6.4.2	Creación y control de pedidos en tiempo real.....	55
6.4.3	Base de datos de clientes.....	56
6.4.4	Arqueo de caja	57
6.4.5	Asignación y gestión de roles	57
6.4.6	Módulo de informes y análisis de datos.....	58
6.4.7	Requerimientos no funcionales.....	58
6.4.8	Requerimientos técnicos	59
6.5	Diagramas de caso de uso	59
6.6	Diagrama entidad/relación	63
6.7	Diagrama de clases	66
6.8	Diagrama de estados	69

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

6.8.1 Relacionados al tenant	69
6.8.2 Funciones básicas del sistema.....	74
6.9 Diagrama de actividades	82
6.9.1 Creación de pedido	82
6.9.2 Edición de pedido en tiempo real.....	83
6.9.3 Confirmación de entrega de pedido	84
6.9.4 Arqueo de caja	85
6.9.5 Gestión de roles y permisos	86
6.9.6 Registro de cliente.....	87
7. Diseño de aplicación.....	88
7.1 Arquitectura planteada	88
7.2 Backend.....	91
7.2.1 Arquitectura de microservicios	92
7.2.1.1 Microservicio de tenants.	92
7.2.1.2 Microservicio de mensajería.	93
7.2.1.3 Servicio proxy de cliente.	93
7.2.2 Protocolo de comunicación.....	93
7.2.3 Tipos de peticiones HTTP soportadas	94
7.2.4 Manejo de parámetros y autenticación	95
7.2.5 Arquitectura basada en componentes.....	96
7.2.5.1 Capa de controladores y rutas.	96
7.2.5.2 Capa de servicios y funciones.....	98
7.2.5.3 Capa de repositorios y clientes HTTP.	99

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

7.2.5.4 Capas adicionales de la arquitectura	101
7.2.6 Gestión de base de datos	101
7.2.6.1 Base de datos de tenants.	102
7.2.6.2 Base de datos de mensajería.	102
7.2.7 Integridad y constraints.....	103
7.2.8 Aislamiento multitenant.....	104
7.2.9 Manejo de transacciones y consistencia	104
7.3 Frontend	106
7.3.1 Arquitectura dual del frontend.....	107
7.3.1.1 Frontend de gestión de tenants.....	107
7.3.1.2 Funcionalidades principales.....	107
7.3.2 Arquitectura y tecnologías	107
7.3.3 Sistema de roles y permisos.....	108
7.3.3.1 Rol de administrador.....	108
7.3.3.2 Rol de operador.....	110
7.3.3.3 Rol de mensajero.....	110
7.3.4 Funcionalidades principales.....	111
7.3.4.1 Autenticación y gestión de sesiones.	111
7.3.4.2 Gestión de empresas de mensajería (tenants).	111
7.3.4.3 Gestión de clientes.	112
7.3.4.4 Gestión de pedidos.....	112
7.3.4.5 Gestión de usuarios.....	113
7.3.4.6 Gestión de mensajeros.	114

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

7.3.4.7 Gestión de arqueos de caja.....	114
7.3.4.8 Dashboard y estadísticas.....	115
7.3.4.9 Gestión de tarifas.....	115
7.3.4.10 Gestión de notificaciones.....	116
7.3.4.11 Interfaz responsiva y experiencia de usuario.....	116
7.3.5 Estructura de componentes.....	116
7.3.5.1 Componentes del frontend de tenants.....	117
7.3.5.5 Componentes del frontend de mensajería.....	117
7.3.6 Integración con el backend.....	118
7.3.6.1 Arquitectura de comunicación.....	118
7.3.6.1.1 Frontend de mensajería proxy-service.....	118
7.3.6.1.2 Frontend de tenants comunicación directa.....	119
7.3.6.2 Sistema de validación y seguridad.....	119
7.3.6.2.1 Validación de tenants.....	119
7.3.6.2.2 Control de accesos.....	120
7.3.6.3 Manejo de peticiones HTTP.....	120
7.3.6.3.1 Configuración de axios.....	120
7.3.6.3.2 Manejo de errores.....	121
7.3.6.3.3 Consistencia de datos.....	121
7.4 Uso de datos.....	122
7.4.1 Arquitectura multitenant.....	122
7.4.1.1 Esquema tenant.....	122
7.4.1.2 Esquema mensajería.....	122

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

7.4.2 Procesamiento de datos.....	123
7.4.2.1 Fase de perfilamiento general.....	123
7.4.2.2 Fase de limpieza de datos.....	124
7.4.2.3 Fase de validación regla de negocios.....	125
7.4.2.3.1 Reglas de tenants y empresas.....	125
7.4.2.3.2 Reglas para usuarios.....	125
7.4.2.3.3 Reglas para clientes.....	126
7.4.2.3.4 Reglas para pedidos.....	126
7.4.2.3.5 Reglas para mensajeros.....	126
7.4.2.3.6 Reglas para direcciones.....	127
7.4.2.3.7 Reglas para tarifas.....	127
7.4.2.3.8 Reglas para arqueos de caja.....	127
7.4.2.3.9 Reglas para historial de pedido.....	128
7.4.3 Validación y persistencia.....	128
7.4.3.1 Validación a nivel de base de datos.....	128
7.4.3.2 Validación a nivel de aplicación.....	128
7.4.3.3 Tipos de datos soportados.....	129
7.4.4 Triggers y automatización.....	129
7.4.4.1 Gestión de mensajeros.....	129
7.4.4.2 Gestión de pedidos.....	129
7.4.4.3 Gestión de arqueos de caja.....	129
7.4.4.4 Gestión de clientes.....	130
7.4.5 Índices y optimización.....	130

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

7.4.5.1 Índices por tenant.....	130
7.4.5.2 Índices operacionales.....	130
7.4.5.3 Índices de rendimiento.....	130
7.4.6 Vistas y reportes.....	131
7.4.6.1 Vistas de estadísticas.....	131
7.4.6.2 Vistas de gestión.....	131
8. Implementación del prototipo.....	131
8.1 Backend Spring Boot.....	132
8.1.1 Dependencias principales.....	132
8.1.2 Autenticación y autorización.....	133
8.1.3 Documentación de apis.....	133
8.1.2 Generación de reportes.....	133
8.2 Frontend React-Vite.....	134
8.2.1 Herramientas de desarrollo.....	134
8.2.2 Comunicación HTTP.....	134
8.2.3 Interfaz de usuario.....	134
8.2.4 Navegación y enrutamiento.....	135
8.2.5 Visualización de datos.....	135
8.2.6 Mapas interactivos.....	135
8.2.7 Autenticación en frontend.....	136
8.3 Base de datos MySQL.....	136
8.3.1 Características principales.....	137
8.4 Contenedor Docker.....	137

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

8.5	Proxy Node.js.....	138
8.6	GitHub.....	139
8.7	Conclusiones del stack tecnológico	139
9.	Validación y pruebas funcionales	140
9.1	Plan de pruebas	140
9.1.1	CP-001: Creación de pedido	140
9.1.2	CP-002: Asignación de mensajero a pedido	141
9.1.3	CP-003: Registro de nuevo cliente	142
9.1.4	CP-004: Creación de arqueo de caja.....	143
9.1.5	CP-005: Edición de información mensajero	143
9.1.6	CP-006: Búsqueda y filtrado de pedidos	144
9.1.7	CP-007: Generación de reporte de ingresos.....	145
9.1.8	CP-008: Actualización de estado pedido	145
9.1.9	CP-009: Configuración de tarifas	146
9.1.10	CP-010: Cierre de arqueo de caja	147
9.1.11	CP-011: Visualización de dashboard con métricas.....	148
9.1.12	CP-012: Gestión de notificaciones.....	148
9.1.13	CP-013: Creación de pedido con datos inválidos	149
9.1.14	CP-014: Creación de pedido anterior de un cliente registrado	150
9.1.15	CP-015: Creación cuarto arqueo de caja en el mismo día	151
9.1.16	CP-016: Asignación de descuento a cliente frecuente.....	151
9.1.17	CP-017: Registro de ingreso manual en arqueo de caja	152
9.1.18	CP-018: Creación de tenant y verificación de aislamiento de datos.....	153

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

9.1.19 CP-019: Visualización de ruta de un pedido en mapa	154
9.1.20 CP-020: Creación de cliente con pedidos anteriores	155
9.1.21 CP-021: Tenant inactivo	156
9.2 Ejecución de las pruebas	157
9.3 Resultados validación	171
10. Conclusiones y trabajo futuro	172
11. Recomendaciones	175
Referencias Bibliográficas	177
Apéndices.....	182

Lista de Tablas

	Pág.
Tabla 1 <i>Estado del arte</i>	38
Tabla 2 <i>Resultados de las pruebas de validación</i>	171

Lista de Figuras

	Pág.
Figura 1 <i>Esquema de metodología de trabajo</i>	42
Figura 2 <i>Diagrama de casos de uso administrador y operador.</i>	60
Figura 3 <i>Diagrama de casos de uso usuarios.</i>	61
Figura 4 <i>Diagrama de casos de uso administrador y operador.</i>	62
Figura 5 <i>Diagrama entidad/relación tenant.</i>	64
Figura 6 <i>Diagrama entidad/relación mensajería.</i>	64
Figura 7 <i>Diagrama de clases tenant.</i>	66
Figura 8 <i>Diagrama de clases mensajería.</i>	67
Figura 9 <i>Diagrama del ciclo de vida del tenant.</i>	69
Figura 10 <i>Diagrama configuración inicial del tenant.</i>	70
Figura 11 <i>Diagrama de gestión de inquilinos del tenant.</i>	70
Figura 12 <i>Diagrama de vida de un usuario del tenant.</i>	71
Figura 13 <i>Diagrama de análisis de datos del tenant.</i>	72
Figura 14 <i>Diagrama de gestión de suscripciones del tenant.</i>	73
Figura 15 <i>Diagrama de estados de un pedido.</i>	74
Figura 16 <i>Diagrama de edición de un pedido en tiempo real.</i>	75
Figura 17 <i>Diagrama de arqueo de caja.</i>	76
Figura 18 <i>Diagrama de gestión de roles y permisos.</i>	77
Figura 19 <i>Diagrama de registro cliente.</i>	78
Figura 20 <i>Diagrama de reasignación de pedidos.</i>	79

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Figura 21	<i>Diagrama de gestión de notificaciones.</i>	80
Figura 22	<i>Diagrama del mantenimiento del sistema</i>	81
Figura 23	<i>Diagrama de actividad creación de pedidos</i>	83
Figura 24	<i>Diagrama de actividad edición de pedidos en tiempo real</i>	83
Figura 25	<i>Diagrama de actividad confirmación de entrega de pedido</i>	84
Figura 26	<i>Diagrama de actividad arqueo de caja</i>	85
Figura 27	<i>Diagrama de actividad gestión de roles y permisos</i>	86
Figura 28	<i>Diagrama de actividad registro del cliente (datos existentes)</i>	87
Figura 29	<i>Diagrama de actividad registro del cliente (nuevos datos)</i>	88
Figura 30	<i>Diagrama de arquitectura final</i>	88
Figura 31	<i>Distribución de pedidos gráfica torta</i>	135
Figura 32	<i>Mapas interactivos</i>	136
Figura 33	<i>Contenedor docker</i>	137
Figura 34	<i>Función del proxy que valida datos del tenant</i>	138
Figura 35	<i>Repositorio github</i>	139
Figura 36	<i>Creación arqueo de caja</i>	158
Figura 37	<i>Verificación correcta creación arqueo de caja</i>	159
Figura 38	<i>Verificación existencia pedido entregado</i>	159
Figura 39	<i>Creación con funcionalidad de pedidos recientes</i>	160
Figura 40	<i>Formulario pedido creación rápida</i>	160
Figura 41	<i>Filtro de búsqueda pedidos</i>	161
Figura 42	<i>Creación de tenant</i>	162
Figura 43	<i>Creación de administrador de mensajería</i>	162

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Figura 44	<i>Contraseña enviada al administrador creado</i>	163
Figura 45	<i>Asociación administrador al tenant</i>	163
Figura 46	<i>Verificación actualización administrador</i>	164
Figura 47	<i>Inicio de sesión con administrador creado</i>	164
Figura 48	<i>Verificación aislamiento de datos en clientes</i>	165
Figura 49	<i>Verificación aislamiento de datos en pedidos</i>	165
Figura 50	<i>Verificación aislamiento de datos en usuarios</i>	166
Figura 51	<i>Verificación de pedido con direcciones correctas</i>	166
Figura 52	<i>Mapa generado por pedido</i>	167
Figura 53	<i>Verificación de pedido sin cliente</i>	168
Figura 54	<i>Formulario de creación de cliente</i>	168
Figura 55	<i>Carga de pedidos sin cliente</i>	169
Figura 56	<i>Formulario de direcciones automatizadas</i>	170
Figura 57	<i>Verificación creación cliente</i>	170

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Lista de Apéndices

	pág.
Apéndice A. Enlace al repositorio donde se encuentra todo el código trabajo.....	182
Apéndice B. Enlace al repositorio donde se encuentran todos los recursos digitales.....	182
Apéndice C. Conjunto de videos demostrativos del prototipo implementado.	182
Apéndice D. Listado completo de casos de uso desarrollados.	182

Glosario

Aplicaciones Web SPA (Single Page Applications): aplicaciones que se cargan en el navegador y no recargan la página durante su uso, ejecutándose como una única página web donde la lógica de presentación reside en el cliente.

Arquitectura de software: nivel de diseño que surge en las primeras etapas del proceso de creación de software, abordando cuestiones estructurales como la organización general del software, estructura de control global, protocolos de comunicación y distribución física de componentes.

Arquitectura multitenant: principio de arquitectura de software que permite que varios clientes (tenants) compartan la misma aplicación e instancia, aprovechando al máximo la economía de escalabilidad mientras mantienen aislamiento de datos.

Instancia única con base de datos compartida: modelo multitenant donde una única instancia de aplicación y base de datos son compartidas por todos los tenants, diferenciando datos mediante un identificador único (tenant_id).

Instancia única con bases de datos separadas: enfoque multitenant que mantiene una aplicación compartida mientras asigna una base de datos independiente a cada tenant.

Instancias y bases de datos completamente aisladas: modelo donde cada tenant opera con una instancia dedicada de aplicación y su propia base de datos independiente.

Mensajería express: servicio especializado de transporte rápido de documentos y paquetes dentro de la misma ciudad o región, caracterizado por plazos de entrega cortos y servicios adicionales como encargos específicos.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

PYMES: pequeñas y medianas empresas que, por su naturaleza flexible y conocimiento local, demuestran capacidad excepcional para adaptarse a las demandas del mercado.

SaaS (Software as a Service): modelo de distribución de software donde las aplicaciones se entregan a través de una red desde un centro de datos centralizado, ofreciendo acceso mediante tarifa recurrente o suscripción.

Tenant: cliente o inquilino en una arquitectura multitenant que utiliza una instancia compartida de software manteniendo sus datos aislados de otros usuarios.

Transformación digital: proceso mediante el cual las organizaciones integran tecnología en sus procesos empresariales para mejorar eficiencia, competitividad y capacidades operativas.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Resumen

Título: Diseño de un sistema software multitenant para apoyar el negocio de mensajería express dirigido a pequeñas y medianas empresas *

Autor: Dilan Esteban Rey Sepúlveda y Silvia Alejandra Cárdenas Santos **

Palabras Clave: Arquitectura multitenant, Mensajería express, Digitalización empresarial, PyMES, Transformación digital, SaaS (Software as a Service).

Descripción:

Este trabajo presenta el desarrollo e implementación exitosa de una solución tecnológica integral basada en arquitectura multitenant, específicamente diseñada para la digitalización de pequeñas y medianas empresas del sector de mensajería express. La investigación aborda la necesidad imperante de modernización tecnológica en un sector dominado por grandes corporaciones como Rappi, que han relegado a las empresas tradicionales mediante infraestructuras tecnológicas avanzadas. La solución desarrollada incluye un sistema completo de gestión operativa que comprende: arcos de caja con reportes detallados, dashboard con estadísticas del negocio y métricas de rendimiento, gestión integral de clientes y personal, administración del ciclo de vida de pedidos con visualización de rutas, configuración de tarifas, administración de usuarios con roles específicos, y centro de notificaciones contextualizado. La arquitectura multitenant permite que múltiples empresas compartan la misma infraestructura manteniendo el aislamiento completo de datos, ofreciendo una alternativa económica y escalable para la digitalización empresarial. Los resultados demuestran que es posible desarrollar herramientas tecnológicas robustas y económicamente accesibles que permiten a las pymes de mensajería competir efectivamente con grandes corporaciones del sector, contribuyendo a la supervivencia y crecimiento de un ecosistema empresarial vital para la economía contemporánea.

* Trabajo de Grado

** Facultad de Ingenierías Fisicomecánicas. Escuela de Ingeniería de Sistemas e Informática. Director Gabriel Rodrigo Pedraza Ferreira. Codirector Jathinson Meneses Mendoza

Abstract

Title: Design of a multi-tenant software system to support the express courier business targeting small and medium-sized enterprises *

Author(s): Dilan Esteban Rey Sepúlveda y Silvia Alejandra Cárdenas Santos **

Key Words: Multitenant architecture, Express Courier, Business digitalization, SMEs, Digital transformation, SaaS (Software as a Service).

Description:

This work presents the successful development and implementation of a comprehensive technological solution based on multitenant architecture, specifically designed for the digitalization of small and medium-sized enterprises in the express courier sector. The research addresses the urgent need for technological modernization in a sector dominated by large corporations like Rappi, which have relegated traditional companies through advanced technological infrastructures. The developed solution includes a complete operational management system comprising: cash reconciliation with detailed reports, business dashboard with statistics and performance metrics, comprehensive client and staff management, order lifecycle administration with route visualization, pricing configuration, user administration with specific roles, and contextualized notification center. The multitenant architecture allows multiple companies to share the same infrastructure while maintaining complete data isolation, offering an economical and scalable alternative for business digitalization. Results demonstrate that it is possible to develop robust and economically accessible technological tools that enable courier SMEs to compete effectively with large sector corporations, contributing to the survival and growth of a business ecosystem vital to the contemporary economy.

* Degree Work

** Faculty of Physical-Mechanical Engineering. School of Systems and Computer Engineering. Director Gabriel Rodrigo Pedraza Ferreira. Co-director Jathinson Meneses Mendoza.

Introducción

En la actualidad, la integración de la tecnología en los procesos empresariales se ha convertido en una necesidad imperante para cualquier organización que aspire a crecer y mantenerse competitiva en un mercado en constante evolución. Las empresas, independientemente de su tamaño o sector, se enfrentan a un entorno donde la digitalización no solo facilita la operación diaria, sino que también redefine las dinámicas de interacción con los clientes y la gestión interna. Adaptarse a esta nueva realidad tecnológica es crucial, ya que no solo proporciona una ventaja competitiva, sino que también permite a las empresas ofrecer servicios más eficientes y personalizados. Entre los beneficios más destacados de esta transformación se encuentran la mejora de la visibilidad del negocio, la optimización de la trazabilidad de los productos y servicios, la flexibilidad en la operación y un alcance ampliado que abre nuevas oportunidades de mercado.

El sector de la mensajería express se ha consolidado como uno de los motores económicos más dinámicos de la actualidad, generando millones de empleos directos e indirectos a nivel mundial y constituyendo un pilar fundamental para el funcionamiento del comercio contemporáneo. Este ecosistema empresarial se organiza principalmente a través de pequeñas y medianas empresas (pymes) quienes, por su naturaleza flexible y conocimiento local, han demostrado una capacidad excepcional para adaptarse a las demandas del mercado y proporcionar soluciones de logística que conectan a productores, comerciantes y consumidores de manera eficiente. La importancia económica de este sector trasciende la simple entrega de

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

paquetes, ya que facilita el crecimiento del comercio electrónico, apoya a emprendedores locales y genera cadenas de valor que benefician a múltiples sectores de la economía. En este contexto, estas pymes del sector mensajería poseen un potencial significativo para beneficiarse de la integración tecnológica, que podría potenciar sus capacidades operativas y competitivas en un mercado cada vez más digitalizado.

Esta necesidad de modernización es particularmente urgente en sectores como el de la mensajería express, que incluye tanto la entrega de paquetería como el servicio de domicilios de comida. En un contexto global donde la demanda por servicios rápidos y eficientes sigue creciendo, las pequeñas y medianas empresas de mensajería se encuentran en una encrucijada. Grandes corporaciones como Rappi han implementado infraestructuras tecnológicas avanzadas y han desarrollado procesos meticulosamente estructurados que han elevado sus niveles de éxito y satisfacción del cliente. Este avance ha tenido un impacto significativo en el mercado, relegando a muchas de estas empresas que han ofrecido estos servicios durante décadas. Sin la adopción de herramientas digitales y estrategias innovadoras, estas empresas corren el riesgo de quedar obsoletas, luchando por encontrar su lugar en un ecosistema dominado por actores consolidados y tecnológicamente avanzados.

La transformación digital, por tanto, no es solo una opción, sino una cuestión de supervivencia. Las pequeñas y medianas empresas de mensajería express deben replantear sus modelos de negocio, explorar nuevas tecnologías y adaptarse a las expectativas cambiantes de los consumidores para poder competir efectivamente y recuperar su relevancia en el mercado. En este contexto, es vital analizar las estrategias que pueden implementar estas empresas para

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

modernizar sus operaciones y aprovechar al máximo las oportunidades que ofrece la tecnología, en este caso, siendo el uso de aplicaciones software ofrecidas como servicio SaaS, técnicamente desarrolladas bajo arquitectura de software multitenant.

Bajo esta premisa se desarrolló e implementó exitosamente una solución tecnológica integral basada en una arquitectura multitenant, diseñada específicamente para satisfacer las necesidades operativas de empresas de mensajería express. La implementación abarcó el desarrollo de varios servicios backend, un proxy para la gestión de peticiones y diversas interfaces frontend, utilizando una combinación de tecnologías modernas que permitieron ofrecer una solución eficiente y con una experiencia de usuario mejorada. Esta solución multitenant permite que múltiples empresas compartan la misma infraestructura tecnológica manteniendo sus datos y procesos completamente aislados, lo que representa una alternativa económica y escalable para empresas que buscan digitalizarse sin incurrir en costos prohibitivos de desarrollo individual.

El sistema desarrollado incluye un conjunto completo de funcionalidades especializadas para la gestión integral de operaciones de mensajería: arqueos de caja con capacidad para generar reportes detallados de ingresos y egresos, un dashboard general con estadísticas comprehensivas del negocio que incluye métricas de pedidos y rendimiento de mensajeros presentadas mediante gráficos interactivos, gestión completa de clientes con funcionalidades de descuentos y creación automática desde pedidos, administración del personal de mensajería incluyendo control de vehículos y disponibilidad, gestión integral del ciclo de vida de pedidos con historial de cambios y visualización de rutas, configuración del sistema de tarifas con opciones de activación y desactivación, administración de usuarios con asignación de roles específicos, centro de notificaciones contextualizado por rol, y gestión de perfiles personalizados. Estas

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

funcionalidades fueron diseñadas para abordar las necesidades específicas identificadas en el sector de mensajería express, proporcionando herramientas que permiten una gestión eficiente y un control operativo completo.

De esta forma, la implementación de esta arquitectura multitenant representa una solución viable y efectiva para la digitalización de pequeñas y medianas empresas de mensajería express, demostrando que es posible desarrollar herramientas tecnológicas robustas y económicamente accesibles que permiten a estas organizaciones competir en igualdad de condiciones con grandes corporaciones del sector.

1. Planteamiento y justificación del problema

Las actividades de mensajería express representan un sector significativo dentro del mercado laboral colombiano. En ciudades como Ibagué, se registran 344 pequeñas y medianas empresas dedicadas a esta actividad económica (Según la Cámara de Comercio de Ibagué con datos tomados a corte del 24 de julio de 2024 (1)), mientras que en Bucaramanga se cuentan 345 (Según la Cámara de Comercio de Bucaramanga con datos tomados a corte del 3 de julio de 2024 (2)). A pesar de este amplio panorama, muchas de estas empresas enfrentan un reto crucial: la falta de infraestructura tecnológica adecuada para competir eficazmente con los grandes actores del sector. Este déficit se traduce en la dependencia de herramientas ineficientes como hojas de cálculo y sistemas basados en papel, que limitan la trazabilidad, el control de domiciliarios y la gestión de entregas. En un entorno donde la agilidad y la eficiencia son esenciales, estas prácticas obsoletas no solo afectan la calidad del servicio, sino que también ponen en riesgo la viabilidad del negocio.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Para abordar este desafío, se propone el desarrollo de un sistema de software con una arquitectura multitenant. Este enfoque permite que múltiples empresas compartan la misma infraestructura y recursos, lo que resulta en una reducción significativa de los costos operativos y de mantenimiento en comparación con la implementación de soluciones tecnológicas individuales. Las pequeñas y medianas empresas no pueden permitirse el lujo de invertir en software personalizado, que suele ser costoso y difícil de mantener. Por lo tanto, el modelo de Software como Servicio (SaaS) se presenta como una solución ideal, ya que ofrece una economía de escala que permite a estas empresas acceder a tecnología avanzada sin la necesidad de realizar grandes inversiones iniciales.

El modelo SaaS no solo proporciona flexibilidad y escalabilidad, sino que también permite la actualización constante del software, asegurando que las empresas siempre cuenten con las últimas funcionalidades y mejoras. Este sistema puede diseñarse para atender las necesidades específicas del sector de mensajería express, facilitando la gestión de entregas, el control de domiciliarios y la trazabilidad de pedidos de manera eficiente. Además, este tipo de soluciones ya ha demostrado su efectividad en otras industrias, como el sector de los restaurantes, donde se han implementado sistemas similares que han optimizado los procesos operativos y mejorado la satisfacción del cliente. Al ofrecer una plataforma sencilla y eficaz, este sistema de software permitirá a las pequeñas y medianas empresas de mensajería digitalizar sus procesos de negocio, mejorando no solo la calidad del servicio, sino también la eficiencia en la gestión interna, lo que les permitirá competir de manera más efectiva en un mercado cada vez más tecnológico y exigente.

2. Objetivos

2.1 Objetivo General

Diseñar un sistema de software multitenant que brinde soporte al negocio de mensajería express, facilitando la transformación digital de pequeñas y medianas empresas dedicadas a este sector.

2.2 Objetivos Específicos

Identificar las características, propiedades y funcionamiento de las aplicaciones que implementan arquitectura software de tipo multitenant.

Identificar los requerimientos asociados a los procesos de negocio de las pequeñas y medianas empresas del sector de la mensajería express.

Diseñar la arquitectura solución (multitenant) para el sistema software que soporte los requerimientos de negocio identificados.

Implementar un prototipo software de aplicación multitenant en base al diseño previamente establecido, que implemente un subconjunto de los requerimientos de negocio identificados.

Validar el prototipo software a través de un caso de uso consolidado, propio del sector de la mensajería express.

3. Alcance

El sistema visionado pretende abordar las siguientes funcionalidades dentro del alcance de su implementación.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

1. Diseñar e implementar un módulo central de mensajería que permita a las empresas del sector gestionar la información de personal, asignar roles, controlar pedidos en tiempo real y llevar control de sus finanzas.
 - a. Diseñar e implementar una base de datos consolidada del personal operativo.
 - b. Implementar funcionalidades para la asignación y gestión de roles de usuarios operativos.
 - c. Implementar una interfaz intuitiva que facilite la creación y control de pedidos en tiempo real, con el fin de llevar un registro detallado de cada pedido desde su recepción hasta su entrega y que tenga en cuenta aspectos como asignación de mensajeros, edición de pedido y control de costos/precios.
 - d. Consolidar base de datos de clientes que apoye en términos de agilidad el proceso de creación de nuevos pedidos.
 - e. Implementar módulo de arqueo de caja para registrar y controlar los ingresos y egresos, y para gestionar las diferentes 'cajas' dentro de la empresa.

2. Integrar un módulo de informes y análisis de datos que apoye a la gerencia en la toma de decisiones mediante reportes gráficos e históricos.
 - a. Desarrollar un dashboard para la generación de reportes gráficos sobre ingresos, egresos, pedidos y otros indicadores clave.
 - b. Implementar reportes históricos que permitan analizar datos como el historial de pedidos, las ganancias generadas de estos y el desempeño de los mensajeros.

3. Crear un módulo de gestión de servicio que permita la administración de usuarios y roles en función del plan contratado, asegurando la adecuación del acceso a las funcionalidades del software. Este módulo apoyará las dinámicas SaaS con las que se plantea ofrecer el servicio.
 - a. Desarrollar un sistema para la gestión de usuarios y la asignación de roles según el plan adquirido por cada empresa cliente.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

4. Desarrollar una página web que ofrezca el software como servicio, explicando la razón del servicio junto con las funcionalidades y requerimientos tanto técnicos como no técnicos. Pruebas y testeo del prototipo de software desarrollado en alguna empresa de mensajería express que sirva como caso de uso.
 - a. Crear una página web que describa claramente la razón de ser y propósito del servicio, las funcionalidades principales del software, los requerimientos técnicos necesarios (hardware, software, conectividad) y los requerimientos no técnicos (capacitación, recursos humanos, procesos).

5. Pruebas y testeo del prototipo de software desarrollado en alguna empresa de mensajería express que sirva como caso de uso.
 - a. Identificar una empresa consolidada en el mercado de la mensajería express y realizar acercamientos para testear el prototipo en dicha empresa.
 - b. Verificación del comportamiento del software y el uso de las funcionalidades implementadas en el flujo de trabajo real de la empresa identificada.
 - c. Conclusiones en base al funcionamiento del software en la prueba de caso de uso.

4. Marco de referencia

A través de esta esta sección se expondrán los conceptos clave y la base teórica que respaldan el desarrollo de este proyecto, abarcando temas como la arquitectura de software, arquitectura multitenant, el Software como Servicio (SaaS, por sus siglas en inglés), y la mensajería express, entre otros. Además, se presenta un estado del arte que examina los avances relacionados con la implementación de arquitecturas multitenant para modelar procesos de negocio y el uso de tecnologías de la información en el soporte de servicios de mensajería express. Estas herramientas son fundamentales, debido a que proporcionan el contexto necesario para entender los pros y contras que conlleva la implementación de software multitenant en este tipo de negocios.

4.1 Marco conceptual

A continuación, se establecen y presentan las bases teóricas vitales para el entendimiento y desarrollo del proyecto.

4.1.1 Arquitectura de software

Según Reynoso (2004) en *Introducción a la Arquitectura de Software* (3) no existe una definición universalmente estipulada de arquitectura de software, sino que circula una vasta cantidad de definiciones y, de hecho, existen grandes compilaciones de definiciones alternativas o contrapuestas, como la colección disponible en el Software Engineering Institute – SEI (4).

Sin embargo, es importante destacar la definición de David Garlan y Mary Shaw (1994) en *An introduction to software architecture* (5) quienes afirman que la arquitectura de software representa un nivel de diseño que surge en las primeras etapas del proceso de creación de software y aborda cuestiones estructurales que van más allá de los algoritmos y las estructuras de datos. Dichas cuestiones incluyen prestar especial atención a la organización general del software y contar con una estructura de control global; los protocolos de comunicación, sincronización y acceso a datos; la composición y asignación de funcionalidades a elementos de diseño; la distribución física de los componentes; la escalabilidad y el rendimiento; así como la selección de otras alternativas de diseño.

4.1.2 Software como servicio (SaaS)

De acuerdo con Seethamraju (2014) en *Adoption of Software as a Service (SaaS) Enterprise Resource Planning (ERP) Systems in Small and Medium Sized Enterprises (SMEs)* (6) el modelo

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

de Software como Servicio (SaaS) se define como una aplicación o servicio que se distribuye a través de una red desde un centro de datos centralizado, ofreciendo acceso y uso a los usuarios mediante una tarifa recurrente. En este modelo, los usuarios generalmente alquilan las aplicaciones o servicios de un proveedor central, quien suministra una única aplicación basada en un conjunto común de código y definiciones de datos. Este conjunto es consumido por todos los usuarios en cualquier momento, y el servicio se utiliza bajo un esquema de pago por uso o suscripción, de acuerdo con las condiciones negociadas en el contrato. A cambio, los usuarios reciben el servicio estipulado en el acuerdo de nivel de servicio (SLA).

Benlian y Hess (2011) en *Opportunities and risks of software-as-a-service: Findings from a survey of IT executives* (7) afirman que el modelo de software como servicio se puede ver como una evolución del modelo de Proveedor de servicios de aplicación (o ASP según sus siglas en inglés); ya que SaaS se ubica un paso delante de ASP debido a que se basa en el uso de arquitecturas multitenant, práctica por la cual adquiere una escalabilidad mucho más económica.

4.1.3 Arquitectura multitenant

Es un **principio** de arquitectura de software cuyo dominio de acción es el modelo del software como negocio (SaaS) que permite aprovechar al máximo la economía de escalabilidad, ya que varios clientes - tenants - comparten la misma aplicación e instancia de base de datos. Al mismo tiempo, los tenants disfrutan de una aplicación altamente configurable, lo que hace que parezca que la aplicación está desplegada en un servidor dedicado. Las principales ventajas del multitenant son una mayor utilización de los recursos de hardware y una mayor facilidad de mantenimiento, sobre todo en lo que respecta al proceso de despliegue. Estas ventajas se traducen en una reducción de los costos globales de las aplicaciones, por lo que esta tecnología

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

resulta particularmente atractiva para proveer servicios dirigidos a pequeñas y medianas empresas. (Bezemer & Zaidman, 2010) (8).

La implementación de sistemas multitenant puede seguir diferentes estrategias, cada una con características específicas en términos de aislamiento, complejidad y costos.

Instancia única con base de datos compartida

En este modelo, una única instancia de la aplicación y una base de datos son compartidas por todos los tenants. La diferenciación de datos se logra mediante un identificador único (tenant_id) presente en todas las tablas relevantes.

Ventajas:

- Reducción significativa de la complejidad operativa y costos de infraestructura.
- Gestión centralizada de actualizaciones y mantenimiento.
- Optimización en el uso de recursos del servidor.

Desventajas:

- Riesgo potencial de filtración de datos si el aislamiento no se implementa correctamente.
- Limitaciones en la personalización específica por tenant.
- Posibles cuellos de botella en el rendimiento con alto volumen de tenants.

Instancia única con bases de datos separadas

Este enfoque mantiene una aplicación compartida mientras asigna una base de datos independiente a cada tenant.

Ventajas:

- Aislamiento robusto de datos entre organizaciones.
- Flexibilidad para implementar personalizaciones específicas por tenant.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Mejor distribución de la carga de base de datos.

Desventajas:

- Incremento en la complejidad de gestión y administración de múltiples bases de datos.
- Costos operativos más elevados debido a la infraestructura adicional.
- Mayor complejidad en procesos de respaldo y recuperación.

Instancias y bases de datos completamente aisladas

Cada tenant opera con una instancia dedicada de la aplicación y su propia base de datos independientes.

Ventajas:

- Máximo nivel de aislamiento y seguridad de datos.
- Capacidad completa de personalización por tenant.
- Escalabilidad independiente para cada cliente.

Desventajas:

- Complejidad operativa y costos significativamente más altos.
- Gestión individualizada de actualizaciones y mantenimiento.
- Limitaciones en la escalabilidad global del sistema.

Adicional a las ventajas y desventajas relacionadas a cada uno de estos enfoques mencionados, existen también algunas consideraciones adicionales que es necesario tener en cuenta al momento de pensar en escoger alguno de estos, tales como las que indican Chong, Carraro y Wolter (2006) en Multi-tenant data architecture (13):

Consideraciones económicas (Chong, Carraro & Wolter, 2006):

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Las aplicaciones optimizadas para un enfoque compartido tienden a requerir un mayor esfuerzo de desarrollo que las aplicaciones diseñadas usando un enfoque más aislado (debido a la complejidad relativa de desarrollar una arquitectura compartida), lo que resulta en costos iniciales más altos. Sin embargo, debido a que pueden soportar más inquilinos por servidor, sus costos operacionales continuos tienden a ser más bajos.

Consideraciones relativas a la seguridad (Chong, Carraro & Wolter, 2006):

Como tu aplicación almacenará datos sensibles de inquilinos, los clientes potenciales tendrán altas expectativas sobre la seguridad, y tus acuerdos de nivel de servicio (SLAs) necesitarán proporcionar fuertes garantías de seguridad de datos. Un concepto erróneo común sostiene que solo el aislamiento físico puede proporcionar un nivel apropiado de seguridad. De hecho, los datos almacenados usando un enfoque compartido también pueden proporcionar una fuerte seguridad de datos, pero requiere el uso de patrones de diseño más sofisticados.

Consideraciones relativas a los tenants (Chong, Carraro & Wolter, 2006):

El número, naturaleza y necesidades de los inquilinos que esperas servir afectan tu decisión de arquitectura de datos de diferentes maneras.

Consideraciones regulatorias (Chong, Carraro & Wolter, 2006):

Las empresas, organizaciones y gobiernos a menudo están sujetos a leyes regulatorias que pueden afectar sus necesidades de seguridad y almacenamiento de registros.

Consideraciones relativas a las competencias técnicas (Chong, Carraro & Wolter, 2006):

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Diseñar una arquitectura multitenant y de instancia única sigue siendo una habilidad muy nueva, por lo que la experiencia en la materia puede ser difícil de conseguir. Si tus arquitectos y personal de soporte no tienen mucha experiencia construyendo aplicaciones SaaS, necesitarán adquirir el conocimiento necesario, o tendrás que contratar personas que ya lo tengan. En algunos casos, un enfoque más aislado puede permitir a tu personal aprovechar más de su conocimiento existente de desarrollo de software tradicional que lo que permitiría un enfoque más compartido.

4.1.4 Mensajería express

La mensajería express, con especial énfasis en el enfoque urbano, se refiere a un servicio especializado de transporte rápido de documentos y paquetes, en el que se recoge un envío de un usuario en un punto de origen (X) y se entrega en otro punto de destino (Z) dentro de la misma ciudad o región cercana (lo que garantiza plazos de entrega cortos, a menudo en el mismo día o incluso en cuestión de horas). Este servicio es prestado por empresas que sirven de transportadoras y se encargan de todo el proceso de logística de recogida, transporte y entrega de los bienes.

A menudo, este tipo de empresas no solo se enfocan en el transporte de documentos y paquetes, sino también ofrecen servicios adicionales, como realizar encargos específicos en nombre de los clientes. Estos encargos pueden incluir la realización de trámites bancarios, el pago de facturas o la compra de artículos en tiendas, entre otros. Este tipo de servicio suele estar disponible bajo demanda y se caracteriza por su flexibilidad, conveniencia y rapidez, adaptándose a las necesidades de los usuarios urbanos que buscan externalizar tareas cotidianas.

4.1.5 Aplicaciones web SPA

Afortunadamente, el concepto de las aplicaciones web SPA (Single Page Applications) se encuentra bien definido en varios libros de desarrollo web y tecnologías relacionadas.

De acuerdo con Mikowski & Powell (2013) en *Single page web applications: JavaScript end-to-end* (9), un SPA es una aplicación que se carga en el navegador y no recarga la página durante su uso. Como todas las aplicaciones, está pensada para ayudar al usuario a completar una tarea, como «escribir un documento» o «administrar un servidor web».

Ahondando un poco más en cuestiones técnicas: “Una aplicación SPA se ejecuta como una única página web. En este enfoque, la capa de presentación de toda la aplicación se ha construido fuera del servidor y se gestiona desde el navegador” (Scott Jr, 2015, p. 4) (10). Además, “el diseño general del SPA es prácticamente el mismo que el del diseño tradicional. Los cambios clave son los siguientes: no hay actualizaciones completas del navegador, la lógica de presentación reside en el cliente, y las transacciones del servidor pueden ser sólo de datos, dependiendo de su preferencia para la representación de datos” (Scott Jr, 2015, p. 7) (10).

4.2 Estado del arte

Tabla 1

Estado del arte

Ref.	Título	Año	Revista	Contribución
(6)	Adoption of Software as a Service (SaaS) Enterprise Resource Planning (ERP) systems in small and medium sized enterprises (SMEs)	2015	Information Systems Frontiers	Resalta los beneficios económicos que adquieren las SMEs (Small and medium enterprises) al implementar sistemas de planificación de recursos empresariales (ERP) que siguen el modelo SaaS.
(7)	Opportunities and risks of software-as-a-service: Findings from a survey of IT executives	2011	Decision Support Systems	Analiza las percepciones de oportunidades y riesgos del SaaS por parte de ejecutivos de TI, identificando que las amenazas de

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

				seguridad son el factor dominante que influye en las percepciones de riesgo general.
(8)	Multi-tenant SaaS applications: maintenance dream or nightmare?	2010	Proceedings of the Joint ERCIM Workshop on Software Evolution (EVOL) and International Workshop on Principles of Software Evolution (IWPSE)	Analiza los desafíos de mantenimiento en aplicaciones SaaS multitenant, explorando si representan una ventaja o una pesadilla desde la perspectiva del mantenimiento de software.
(9)	Single page web applications: JavaScript end-to-end	2013	Manning Publications	Proporciona una guía completa para el desarrollo de aplicaciones web de página única (SPA), fundamentales en la arquitectura de sistemas SaaS modernos.
(10)	SPA design and architecture: Understanding single-page web applications	2015	Simon and Schuster	Explica los principios de diseño y arquitectura de aplicaciones de página única, relevantes para la implementación de interfaces de usuario en sistemas SaaS.
(11)	Digital business strategy: Toward a next generation of insights	2013	MIS Quarterly	Proporciona un marco teórico sobre cómo las PYMES pueden beneficiarse de un sistema multitenant en el contexto de mensajería express.
(12)	Small and medium enterprises using software as a service: Exploring the different roles of intermediaries	2010	Australasian Journal of Information Systems	Provee información acerca de cómo se puede servir de intermediario (Proveedor SaaS) para dar soporte a los procesos de negocios de SMEs (Small and médium enterprises) mediante el uso de sistemas SaaS. Resalta que el servicio SaaS provee beneficios significantes para las SMEs.
(13)	Multi-tenant data architecture	2006	MSDN Library, Microsoft Corporation	Provee una vista general de todos los aspectos (tanto técnicos como de negocio) a tener en cuenta al momento de diseñar una arquitectura multitenant. Adicionalmente, presenta distintos enfoques para crear la arquitectura multitenant.
(14)	Architecture strategies for catching the long tail (Software-as-a-Service (SaaS))	2006	MSDN Library, Microsoft Corporation	Presenta estrategias arquitectónicas para aprovechar el modelo de "cola larga" en SaaS, permitiendo servir eficientemente a múltiples segmentos de mercado.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

(15)	Software-as-a-Service and Cloud Computing, a solution for small and medium-sized companies	2010	Bulletin of the Transilvania University of Brasov. Series V: Economic Sciences	Demuestra cómo SaaS y cloud computing representan una solución viable y económica para pequeñas y medianas empresas.
(16)	Towards dynamic tenant management for microservice-based multi-tenant SaaS applications	2018	Proceedings of the 11th Innovations in Software Engineering Conference	Propone un enfoque de gestión dinámica de tenants en aplicaciones SaaS multitenant basadas en microservicios, mejorando la flexibilidad y escalabilidad.
(17)	Multi-tenant, secure, load disseminated SaaS architecture	2010	Proceedings of the 12th International Conference on Advanced Communication Technology	Presenta una arquitectura SaaS multitenant que enfatiza la seguridad y la distribución de carga, abordando desafíos críticos en sistemas multitenancy.
(18)	Determinants of software-as-a-service benefits and impact on firm performance	2019	Decision Support Systems	Identifica los factores determinantes que influyen en los beneficios del SaaS y su impacto en el rendimiento empresarial, proporcionando evidencia empírica sobre los resultados organizacionales.
(19)	Software-as-a-service (SaaS): perspectives and challenges	2014	Science China Information Sciences	Ofrece una perspectiva integral sobre SaaS, identificando oportunidades y desafíos técnicos y de negocio en la implementación de estos sistemas.
(20)	Multi-tenant databases for software as a service: Schema-mapping techniques	2008	Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data	Introduce técnicas de mapeo de esquemas para bases de datos multitenant, fundamentales para la eficiencia y aislamiento de datos en sistemas SaaS.
(21)	Customizing Multi-Tenant SaaS by Microservices: A Reference Architecture	2019	2019 IEEE International Conference on Web Services (ICWS)	Presenta un diseño y arquitectura de referencia nativa de la nube para personalizar SaaS multitenant mediante microservicios, logrando un equilibrio entre aislamiento, asimilación y economías de escala en contextos multitenant.
(22)	Towards a multi-tenant microservice architecture: An industrial experience	2022	2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)	Documenta una experiencia industrial en la implementación de arquitecturas de microservicios multitenant, proporcionando lecciones aprendidas y mejores prácticas.
(23)	Adaptive performance	2015	2015 IEEE/ACM	Desarrolla un middleware de

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

	isolation middleware for multi-tenant SaaS		8th International Conference on Utility and Cloud Computing (UCC)	aislamiento de rendimiento adaptativo para sistemas SaaS multitenant, mejorando la calidad de servicio.
(24)	Scalable resource provisioning for multi-tenant SaaS with cloud functions	2022	2022 2nd International Conference on Cloud Computing, Performance and Quality of Service Testing (CCPQT)	Propone un modelo de aprovisionamiento escalable de recursos para SaaS multitenant utilizando funciones en la nube.
(25)	Container based scalability and performance analysis of multitenant SaaS applications	2022	2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT)	Analiza la escalabilidad y rendimiento de aplicaciones SaaS multitenant basadas en contenedores, evaluando su efectividad.
(26)	Using microservices for non-intrusive customization of multi-tenant SaaS	2019	Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering	Demuestra cómo los microservicios pueden facilitar la customización no intrusiva en aplicaciones SaaS multitenant, mejorando la flexibilidad.
(27)	SPL integrated with microservices: A hybrid architectural proposal for multitenant SaaS	2023	Proceedings of the 17th Brazilian Symposium on Software Components, Architectures, and Reuse	Propone una arquitectura híbrida que integra líneas de productos de software (SPL) con microservicios para sistemas SaaS multitenant, combinando reutilización y modularidad.

5. Metodología

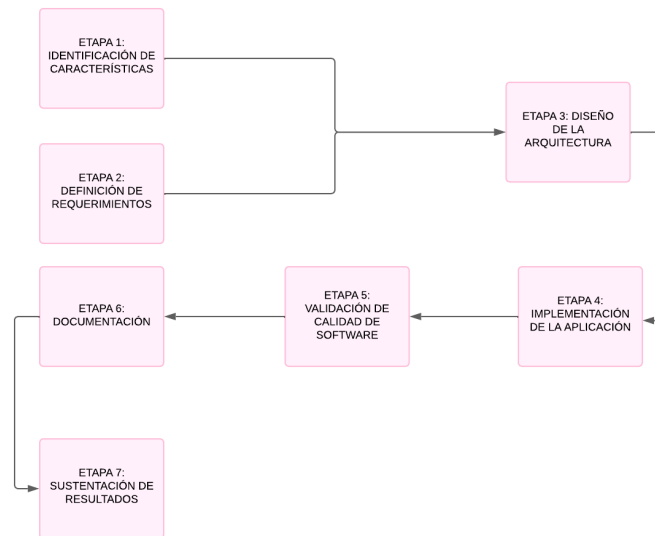
Este apartado expone el enfoque metodológico utilizado en el desarrollo del software multitenant destinado a pequeñas y medianas empresas de mensajería express. Se contempla un proceso integral que abarca desde la identificación de necesidades y características hasta la

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

validación del sistema en entornos reales. Cada etapa incluye actividades específicas y resultados esperados, lo que permite un desarrollo coherente y sistemático, se busca no solo cumplir con los requerimientos del sector, sino también garantizar la usabilidad y el rendimiento del software en su aplicación práctica.

Figura 1

Esquema de metodología de trabajo



5.1 Identificación de características

En esta primera fase se establecen las características clave del sistema que satisfacen las necesidades de las pequeñas y medianas empresas de mensajería express, considerando desde el inicio la arquitectura multitenant para asegurar un funcionamiento eficiente y seguro.

Actividades:

- A.1.1 Análisis de necesidades del cliente.
- A.1.2 Sesiones de lluvia de ideas para generar ideas sobre características del sistema.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- A.1.3 Clasificación y priorización de características considerando como contribuye al entorno multitenant.
- A.1.4 Documentación de características identificadas.

Resultados:

- R.1.1 Conjunto de características identificadas, documentadas y priorizadas que reflejan las necesidades de las pequeñas y medianas empresas de mensajería express, asegurando la alineación con la arquitectura multitenant y sirviendo como base para la fase de definición de requerimientos.

5.2 Definición de requerimientos

En esta etapa se buscan establecer los requisitos funcionales y no funcionales del sistema que respondan efectivamente a las características identificadas en la fase anterior, asegurando que se alineen con las necesidades de las pequeñas y medianas empresas de mensajería express y la arquitectura multitenant.

Actividades:

- A.2.1 Validación de Características Identificadas
- A.2.2 Documentación de requisitos funcionales y no funcionales.
- A.2.3 Priorización de requisitos.

Resultados:

- R.2.1 Conjunto de requisitos funcionales y no funcionales estructurados, validados y priorizados, que servirán de base para el desarrollo del sistema, asegurando la alineación con las características identificadas y las necesidades del entorno multitenant.

5.3 Diseño de la arquitectura

En esta tercera etapa se diseñará la arquitectura del software multitenant sobre los requisitos previamente establecidos. Se desarrollará una estructura técnica que asegure la escalabilidad y el manejo correcto de los datos entre las pequeñas y medianas empresas de mensajería express, permitiendo un funcionamiento eficiente y flexible del sistema, garantizando que se alineen con las expectativas y necesidades identificadas en la fase de definición de requerimientos.

Actividades:

- A.3.1 Evaluación de tecnologías adecuadas para la implementación de la arquitectura multitenant, considerando los requisitos definidos.
- A.3.2 Creación de diagramas de arquitectura que representen la infraestructura y las interacciones del sistema.
- A.3.3 Validación del diseño arquitectónico con los actores clave, asegurando su viabilidad.

Resultados:

- R.3.1 Diagrama de arquitectura multitenant que detalla la estructura del software y sus componentes, alineado con los requerimientos previos.
- R.3.1 Documento técnico que justifica las decisiones de diseño y las tecnologías seleccionadas, garantizando que el sistema cumpla con las necesidades específicas recopiladas en la fase anterior.

5.4 Implementación de la aplicación

En esta fase se procederá al desarrollo de la aplicación multitenant, utilizando la arquitectura diseñada previamente y alineándose con los requisitos funcionales y no funcionales definidos.

Actividades:

- A.4.1 Implementación de los módulos del software según la arquitectura y requisitos establecidos.
- A.4.2 Revisión periódica del progreso del desarrollo y ajustes basados en el feedback recibido.

Resultados:

- R.4.1 Aplicación multitenant funcional con las principales características implementadas, lista para pruebas.

5.5 Validación de calidad de software

En esta etapa se llevará a cabo la evaluación de la calidad del software, centrada en el testeo y la validación de la aplicación multitenant desarrollada. Se implementarán diversas pruebas, incluyendo pruebas funcionales, de rendimiento y de seguridad, para asegurar que el sistema cumple con los requerimientos establecidos y proporciona una experiencia óptima para los usuarios finales.

Actividades:

- A.5.1 Ejecución de pruebas funcionales para verificar que todas las características del software operan correctamente.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- A.5.2 Realización de pruebas de rendimiento para evaluar la escalabilidad y la capacidad de respuesta del sistema.
- A.5.3 Conducción de pruebas de seguridad para identificar y mitigar vulnerabilidades.

Resultados:

- R.5.1 Informe de pruebas que documenta los resultados de las pruebas funcionales, de rendimiento y de seguridad, con recomendaciones para mejoras.

5.6 Documentación

En esta fase se elaborará la documentación del software multitenant, abarcando tanto la documentación técnica como la guía de usuario. Este proceso es fundamental para asegurar que el sistema sea comprensible y accesible para los futuros desarrolladores y usuarios finales. Se incluirán detalles sobre la arquitectura, los módulos implementados y las instrucciones necesarias para la operación del software, facilitando su mantenimiento y actualización a largo plazo.

Actividades:

- A.6.1 Redacción de la documentación técnica que incluya la arquitectura del sistema, especificaciones de diseño y guías de implementación.
- A.6.2 Creación de un manual de usuario que detalle el uso del software, incluyendo ejemplos y procedimientos.
- A.6.3 Revisión y validación de la documentación con usuarios finales para asegurar su claridad y utilidad.

Resultados:

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- R.6.1 Documentación técnica completa que proporciona información detallada sobre la arquitectura y funcionamiento del software.
- R.6.2 Manual de usuario accesible y claro, que facilita la capacitación y el uso efectivo del sistema por parte de los usuarios finales.

5.7 Sustentación de resultados

Como última etapa se llevará a cabo la sustentación de los resultados obtenidos a lo largo del desarrollo del software multitenant. Se presentarán los hallazgos y logros del proyecto ante un comité evaluador, destacando las características del sistema, su funcionalidad y su impacto en las pequeñas y medianas empresas de mensajería express. Este proceso no solo servirá para validar el trabajo realizado, sino también para recibir retroalimentación y sugerencias que puedan enriquecer futuras implementaciones.

Actividades:

- A.7.1 Preparación de una presentación que resuma el desarrollo del proyecto, incluyendo la metodología, el diseño, la implementación y los resultados obtenidos.
- A.7.2 Realización de una demostración en vivo del software para mostrar sus funcionalidades y beneficios.

Resultados:

- R.7.1. Validación del proyecto y reconocimiento de los logros alcanzados, que sirve como base para futuros desarrollos y mejoras del sistema.

6. Análisis y requerimientos del sistema

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

El sistema implementará una arquitectura multitenant, esta aproximación optimiza los recursos compartidos mientras mantiene la separación lógica de datos entre diferentes clientes.

6.1 Enfoque seleccionado

Para el desarrollo del prototipo se ha seleccionado el modelo de **instancia única con base de datos compartida**. Esta decisión se fundamenta en los siguientes criterios:

6.1.1 Justificación técnica

- **Simplicidad de implementación:** Facilita el desarrollo inicial y reduce la complejidad del sistema.
- **Eficiencia de recursos:** Optimiza el uso de infraestructura durante la fase de prototipado.
- **Flexibilidad futura:** Permite migración a arquitecturas más complejas según evolucionen los requisitos.

6.1.2 Implementación específica

La estrategia de implementación incluye los siguientes elementos técnicos:

Identificación de tenants:

- Incorporación del campo `tenant_id` en todas las tablas que contengan datos específicos de organizaciones.
- Implementación de restricciones a nivel de base de datos para garantizar la integridad referencial.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Filtrado de datos:

- Aplicación automática de filtros tenant_id en todas las consultas SQL.
- Implementación de middleware de seguridad que valide el contexto del tenant en cada operación.

Seguridad y aislamiento:

- Validación estricta de permisos basada en el tenant del usuario autenticado.
- Auditoría de accesos para monitorear posibles violaciones de aislamiento.

Esta aproximación proporciona un equilibrio adecuado entre funcionalidad, seguridad y costo para las necesidades actuales del proyecto, manteniendo la posibilidad de evolución arquitectónica en fases posteriores del desarrollo.

6.2 Alcance del sistema (instancia de la aplicación)

1. Módulo central de mensajería que permita gestionar la información de personal, asignar roles, controlar pedidos en tiempo real y llevar control de sus finanzas.
 - a. Base de datos consolidada del personal operativo.
 - b. Asignación y gestión de roles de usuarios operativos.
 - c. Creación y control de pedidos en tiempo real, que tenga en cuenta aspectos como asignación de mensajeros, edición de pedidos y control de costos/precios.
 - d. Base de datos de clientes.
 - e. Submódulo de arqueo de caja que permita registrar y controlar los diferentes flujos de ingresos y egresos.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

2. Módulo de informes y análisis de datos que apoye a la gerencia en la toma de decisiones mediante reportes gráficos e históricos.
 - a. Generación de reportes gráficos sobre ingresos, egresos, pedidos y otros indicadores clave.
 - b. Reportes históricos que permitan analizar datos como el historial de pedidos, la ganancia de estos y el desempeño de los mensajeros.

6.3 Requerimientos del sistema (instancia de la aplicación)

6.3.1 Requerimientos funcionales

1. Módulo central de mensajería
 - a. Gestión de personal
 - i. RF-001: El sistema debe permitir la creación, edición y eliminación de perfiles de personal operativo (por parte de un administrador de mensajería autorizado). Los usuarios operativos que tendrán acceso de forma directa al sistema son: Administrador de mensajería express, Operador de mensajería express y Mensajero de mensajería express (planteado para un desarrollo móvil futuro).
 - ii. RF-002: Debe consolidar una base de datos con información detallada del personal (nombre, contacto, entre otros). Entiéndase como personal: Mensajeros, operadores, administradores, entre otros.
 - iii. RF-003: Debe permitir la búsqueda y filtrado del personal operativo según diferentes criterios.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

2. Creación y control de pedidos en tiempo real

- i. RF-004: El sistema debe permitir la creación de pedidos con detalles como cliente, dirección de origen y destino, costos, estado, asignación de mensajeros, entre otros.
- ii. RF-005: Debe permitir la asignación de mensajeros según disponibilidad.
- iii. RF-006: Los pedidos deben poder ser editados en tiempo real en caso de cambios por parte del cliente o el operador.
- iv. RF-007: Debe registrar el historial de cambios en cada pedido.
- v. RF-008: El sistema debe guardar los costos y precios asociados al servicio (de acuerdo a los valores indicados por parte del usuario que está creando el pedido).
- vi. RF-009: El sistema debe generar notificaciones automáticas sobre la asignación y edición de pedidos para mensajeros, así como notificaciones generales de cambios en pedidos para administradores/operadores. Estas notificaciones pueden enviarse a través de mensajes de texto o aplicaciones de mensajería instantánea como WhatsApp o Telegram, dado que inicialmente no se contempla un módulo propio de interacción del mensajero con el sistema, aunque se considera el desarrollo futuro de una aplicación móvil dedicada para mensajeros.

3. Bases de datos de clientes

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- i. RF-010: El sistema debe permitir almacenar información de clientes, incluyendo nombre, contacto, direcciones comunes, pedidos previos, entre otros.
- ii. RF-011: Debe permitir la búsqueda y filtrado de clientes por diferentes criterios.
- iii. RF-012: Debe permitir la asignación de condiciones especiales para clientes registrados, como descuentos.

4. Arqueo de caja

- i. RF-013: El sistema debe registrar los ingresos y egresos generados por los servicios.
- ii. RF-014: Debe permitir realizar arqueos de caja en cualquier turno y generar reportes detallados de cada arqueo.
- iii. RF-015: Debe permitir clasificar los ingresos y egresos según su tipo (automático o manual).
- iv. RF-016: Debe generar alertas en caso de inconsistencias en los arqueos.

5. Asignación y gestión de roles

- i. RF-017: El sistema debe permitir asignar roles y permisos a los usuarios según su nivel de acceso:
 - Administrador de mensajería: Acceso total a todas las funcionalidades del sistema.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Operador de mensajería: Acceso a base de datos de clientes, creación y control de pedidos en tiempo real, arqueo de caja de forma diaria, gestión de los mensajeros y a tarifas.
- ii. RF-018: Debe garantizar que los usuarios solo accedan a las funcionalidades según su rol asignado.
- iii. RF-019: Debe permitir modificar roles y permisos en cualquier momento por parte de un administrador autorizado.

6. Módulo de informes y análisis de datos

a. Reportes gráficos

- i. RF-020: El sistema debe permitir generar gráficos sobre ingresos en distintos periodos de tiempo (diario, semanal, mensual, anual).
- ii. RF-021: Debe generar reportes visuales sobre el desempeño de los mensajeros (cantidad de pedidos entregados, tiempos promedio de entrega, entre otros).
- iii. RF-022: Debe permitir exportar los reportes en formatos como PDF y XLS.

6.3.2 Requerimientos no funcionales

1. Usabilidad

- i. RNF-001: La interfaz debe ser intuitiva y fácil de usar para los empleados de las empresas. Se recomienda que los usuarios tengan un nivel mínimo de competencia en el uso de herramientas digitales.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

2. Rendimiento

- i. RNF-002: El sistema debe procesar y actualizar pedidos en tiempo real sin retrasos significativos.
- ii. RNF-003: El sistema debe soportar un flujo mínimo y máximo de solicitudes, considerando que se encuentra alojado en un entorno multi-tenant, el cual será responsable de gestionar la cantidad de solicitudes permitidas.

3. Seguridad

- i. RNF-004: El sistema debe contar con autenticación de usuarios mediante credenciales seguras.
- ii. RNF-005: Se debe encriptar la información sensible en la base de datos.

4. Escalabilidad

- i. RNF-006: El sistema debe ser modular y escalable para permitir la integración de nuevas funcionalidades en el futuro.

6.3.2 Requerimientos técnicos

- i. RT-001: El sistema debe ser compatible con distintos navegadores web, priorizando los navegadores modernos sin excluir versiones anteriores que aún sean utilizadas de manera significativa.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- ii. RT-002: El sistema debe ser compatible con una arquitectura multi-tenant, dado que será alojado en un entorno de este tipo. Se opta por una instancia única de la aplicación con base de datos compartida, diferenciando los datos mediante un campo identificador de tenant.

6.4 Historias de usuario (HU)

6.4.1 Gestión de personal

- HU-001: Como administrador mensajería, quiero poder crear, editar y eliminar perfiles de personal operativo para gestionar el acceso y la información del equipo de trabajo.
- HU-002: Como administrador de mensajería, quiero consolidar una base de datos con la información detallada del personal para facilitar la consulta y gestión de los empleados.
- HU-003: Como administrador de mensajería, quiero poder buscar y filtrar el personal operativo según diferentes criterios para agilizar la localización de información relevante.

6.4.2 Creación y control de pedidos en tiempo real

- HU-004: Como administrador de mensajería y/u operador, quiero poder crear pedidos con detalles completos (cliente, dirección de origen y destino, costos, estado, asignación de mensajero, entre otros) para garantizar un servicio eficiente.
- HU-005: Como administrador de mensajería y/u operador, quiero poder asignar mensajeros a pedidos según su disponibilidad para optimizar la entrega.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- HU-006: Como administrador de mensajería y/u operador, quiero poder editar los pedidos en tiempo real para reflejar cambios solicitados por el cliente o que sean requeridos por la operación (por ejemplo, cambio de mensajero/s asignado/s).
- HU-007: Como administrador de mensajería y/u operador, quiero que el sistema registre el historial de cambios de cada pedido para llevar un control detallado.
- HU-008: Como administrador de mensajería y/u operador, quiero que el sistema guarde los costos y precios asociados al servicio para asegurar la correcta facturación.
- HU-009: Como mensajero, quiero recibir notificaciones sobre la creación y modificaciones de pedidos para mantenerme informado en tiempo real.

6.4.3 Base de datos de clientes

- HU-010: Como administrador de mensajería y/u operador, quiero contar con información registrada de clientes para facilitar futuras interacciones y mejorar el servicio.
- HU-011: Como administrador de mensajería y/u operador, quiero poder buscar y filtrar clientes por diferentes criterios para acceder rápidamente a la información.
- HU-012: Como administrador de mensajería, quiero poder asignar condiciones especiales (descuentos) a clientes registrados para incentivar su lealtad.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

6.4.4 Arqueo de caja

- HU-013: Como administrador de mensajería, quiero contar con un registro de los ingresos y egresos generados por los servicios de mensajería prestados, para mantener un control financiero.
- HU-014: Como administrador de mensajería, quiero realizar arqueos de caja en cualquier momento del día valiéndome en turnos asignados.
- HU-015: Como administrador de mensajería, quiero que los operadores puedan realizar arqueos de caja diarios, para asegurar un control financiero adecuado y ajustado a las necesidades operativas.
- HU-016: Como operador, quiero clasificar los ingresos y egresos asociados a servicios de mensajería según su tipo (automático o manual) para un mejor seguimiento financiero.
- HU-017: Como administrador de mensajería, quiero recibir alertas en caso de inconsistencias en los arqueos para detectar errores o posibles fraudes.

6.4.5 Asignación y gestión de roles

- HU-018: Como administrador de mensajería, quiero asignar roles y permisos a los usuarios según su nivel de acceso para mantener la seguridad y organización del sistema.
- HU-019: Como usuario, quiero acceder solo a las funcionalidades permitidas según mi rol asignado para evitar accesos indebidos.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- HU-020: Como administrador de mensajería, quiero poder modificar roles y permisos en cualquier momento para ajustar la gestión de usuarios según las necesidades del negocio.

6.4.6 Módulo de informes y análisis de datos

- HU-021: Como administrador de mensajería, quiero generar gráficos sobre ingresos en distintos periodos para analizar la rentabilidad del negocio.
- HU-022: Como administrador de mensajería, quiero generar reportes visuales sobre el desempeño de los mensajeros para evaluar su eficiencia y presentarlo a los mismos mensajeros.
- HU-023: Como usuario, quiero exportar los reportes en formatos PDF y XLS para facilitar su análisis y presentación.

6.4.7 Requerimientos no funcionales

- HU-024: Como usuario, quiero que la interfaz del sistema sea intuitiva y fácil de usar para agilizar mis tareas diarias.
- HU-025: Como usuario, quiero que el sistema procese y actualice pedidos en tiempo real sin retrasos para brindar un servicio eficiente.
- HU-026: Como usuario, quiero que el sistema cuente con autenticación segura para proteger mi información y la de mis clientes.
- HU-027: Como administrador, quiero que la información sensible en la base de datos esté encriptada para evitar accesos no autorizados.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- HU-028: Como desarrollador, quiero que el sistema sea modular y escalable para facilitar la integración de nuevas funcionalidades en el futuro.

6.4.8 Requerimientos técnicos

- HU-029: Como usuario, quiero que el sistema sea compatible con distintos navegadores web para poder acceder desde ellos sin restricciones.
- HU-030: Como desarrollador, quiero que la instancia de la aplicación sea compatible con una arquitectura multi-tenant para que sea gestionada sin problemas.

6.5 Diagramas de caso de uso

Para la creación del diagrama de casos de uso se identificaron los actores principales y los casos de uso fundamentales que representan las funcionalidades clave del sistema.

Actores:

- **Superadministrador:** Actor encargado de la gestión global del sistema y de los tenants.
- **Administrador:** Responsable de la administración interna de un tenant específico.
- **Operador:** Usuario que ejecuta tareas operativas como la gestión de pedidos o el manejo de clientes.
- **Mensajero:** Encargado de realizar las entregas y confirmar pedidos.

Casos de uso principales:

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Gestión de tenants: Registrar, modificar, suspender, activar y eliminar un tenant.
- Gestión de personal: Crear, editar y eliminar registros de empleados.
- Gestión de pedidos: Crear, asignar, modificar y registrar pedidos en tiempo real.
- Gestión de clientes: Registrar, buscar y gestionar clientes.
- Arqueo de caja: Registrar ingresos y egresos, realizar arqueos y generar reportes financieros.
- Gestión de roles: Asignar y modificar roles de usuario.
- Generación de reportes: Crear reportes y exportarlos en PDF o XLS.

Para facilidad de visualización se adjuntan los diagramas de uso agrupados por funcionalidad:

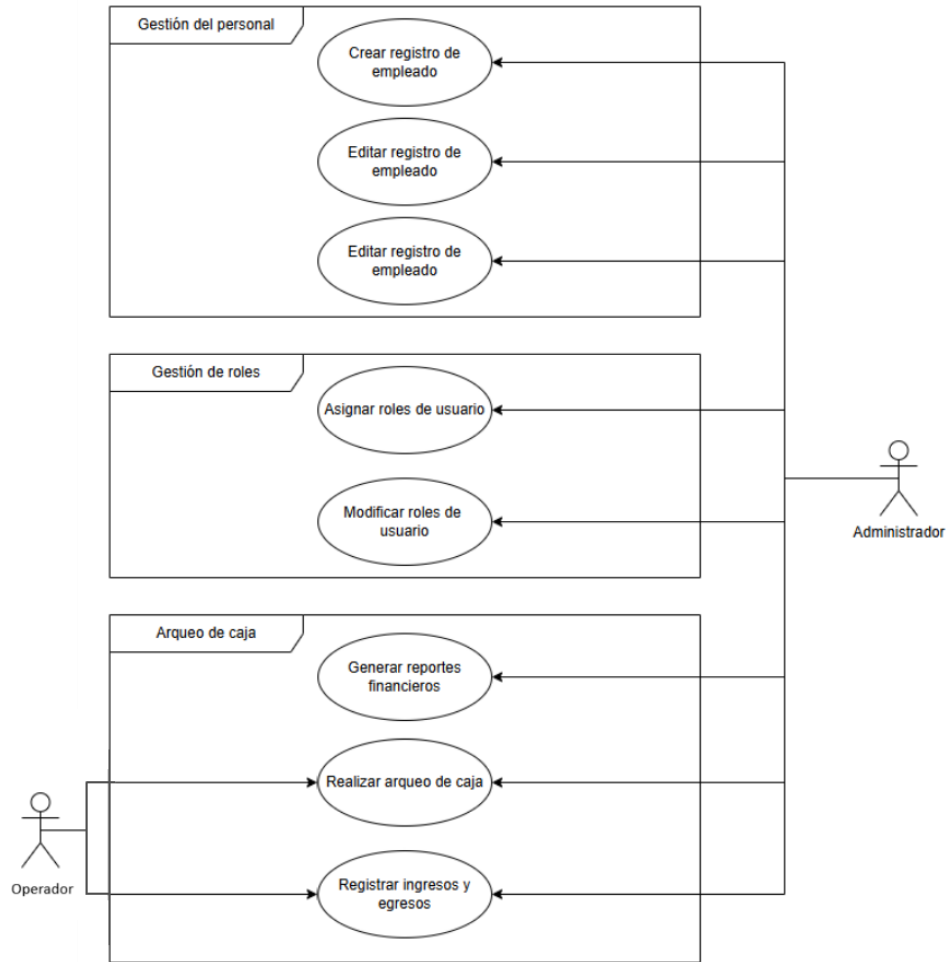
Gestión del personal, gestión de roles y arqueo de caja:

- Actores relacionados: Administrador (de mensajería) y operador.

Figura 2

Diagrama de casos de uso administrador y operador.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



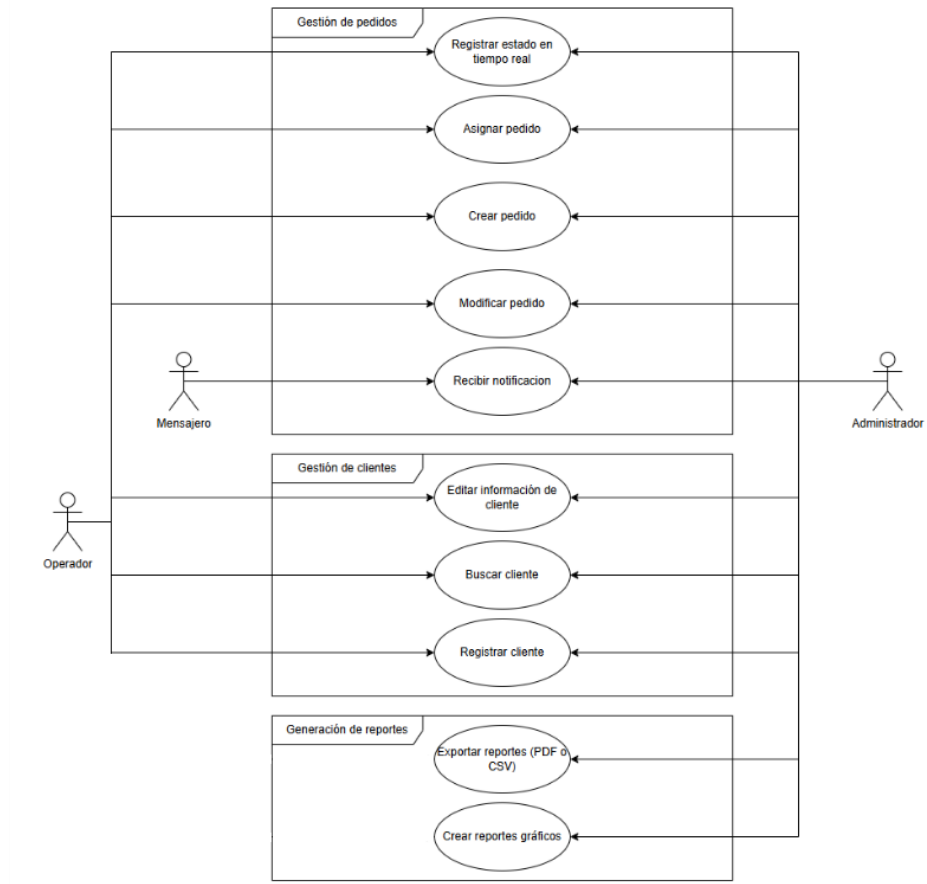
Gestión de pedidos, gestión de clientes y generación de reportes:

- Actores relacionados: Administrador (de mensajería), mensajero, operador.

Figura 3

Diagrama de casos de uso usuarios.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



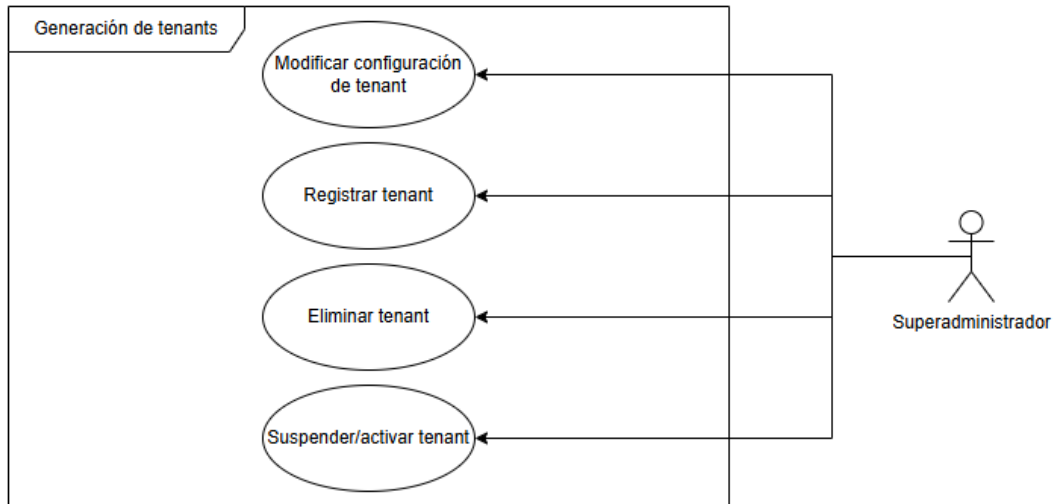
Generación de tenants:

- Actores relacionados: Administrador (del sistema).

Figura 4

Diagrama de casos de uso administrador y operador.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



6.6 Diagrama entidad/relación

Los siguientes diagramas entidad-relación representan la estructura lógica de un sistema de mensajería multiempresa organizado en dos niveles: el **esquema Tenant** que gestiona la administración global de empresas con sus planes y estados, y el **esquema Mensajería** que modela las operaciones específicas de cada empresa. El esquema Mensajería incluye la jerarquía de usuarios (desde administradores hasta mensajeros), la gestión completa de pedidos con sus estados y cambios, el registro de clientes con historial, la auditoría de operaciones, y las entidades relacionadas con la administración financiera (arqueo de caja) y notificaciones. Este modelo de dos capas garantiza la separación segura entre empresas a nivel global, mientras permite operaciones independientes y escalables dentro de cada tenant.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Figura 5

Diagrama entidad/relación tenant.

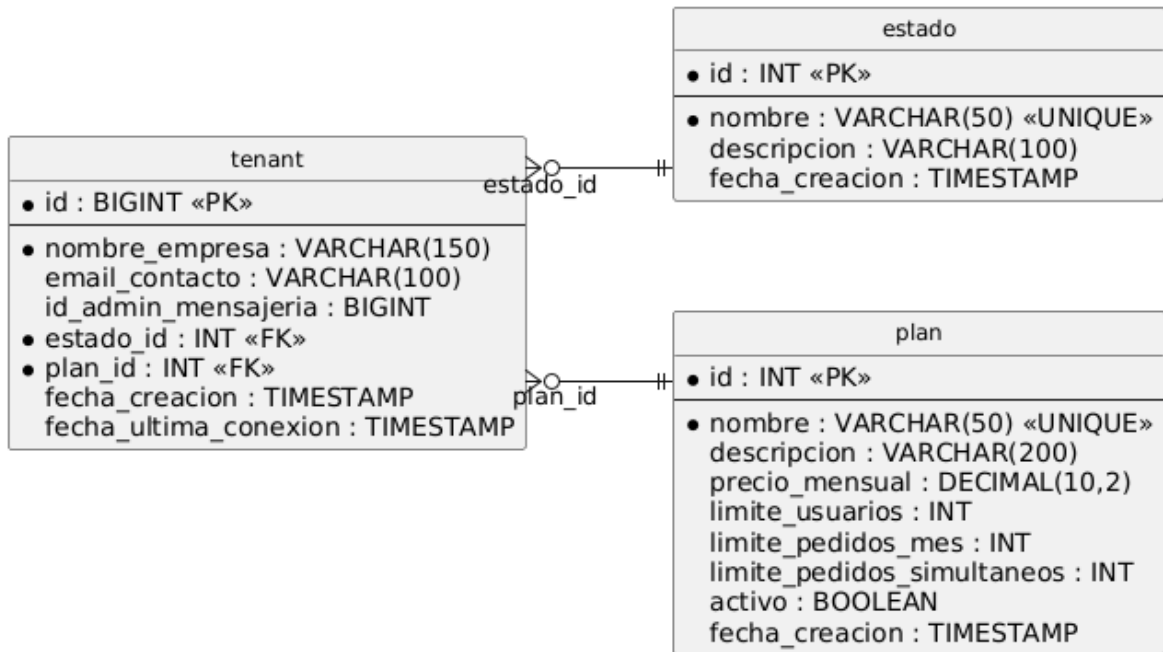
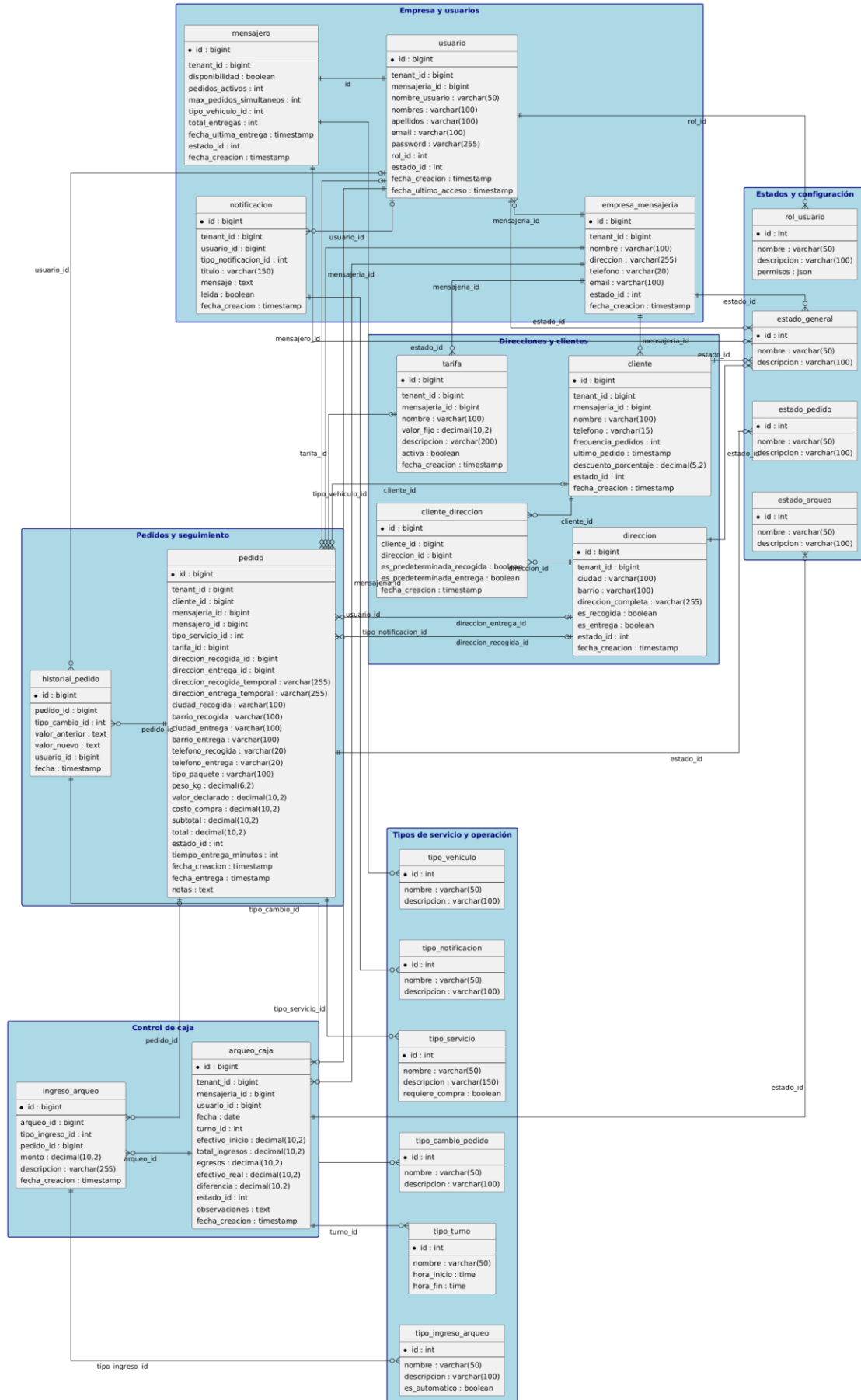


Figura 6

Diagrama entidad/relación mensajería.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



6.7 Diagrama de clases

Los diagramas de clases representan la estructura estática del sistema multi-tenant de mensajería, modelando las principales entidades, sus atributos y relaciones en dos niveles arquitectónicos.

El **diagrama de gestión de tenants** modela la administración centralizada del sistema, incluyendo las entidades **Tenant**, **Plan** y **Estado**, junto con sus servicios y repositorios correspondientes. Se implementa el patrón Repository para el acceso a datos y se utilizan enumeraciones para definir tipos de estados y planes de suscripción.

El **diagrama operativo de mensajería** representa el core del negocio, donde se utilizó especialización para la clase **Usuario** en **Mensajero**, permitiendo comportamientos específicos de entrega. Las entidades principales incluyen **Pedido**, **Cliente**, **Empresa Mensajería** y **Arqueo de Caja**, con múltiples tablas de configuración (estados, tipos de servicio, roles) que proporcionan flexibilidad al sistema.

La arquitectura multi-tenant se implementa mediante el campo `tenant_id` en las entidades operativas, garantizando el aislamiento de datos entre diferentes empresas de mensajería. También se incluye un sistema completo de auditoría a través de **historial de pedidos** y control financiero mediante **arqueos de caja**, facilitando el diseño orientado a objetos y la escalabilidad del sistema.

Figura 7

Diagrama de clases tenant.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

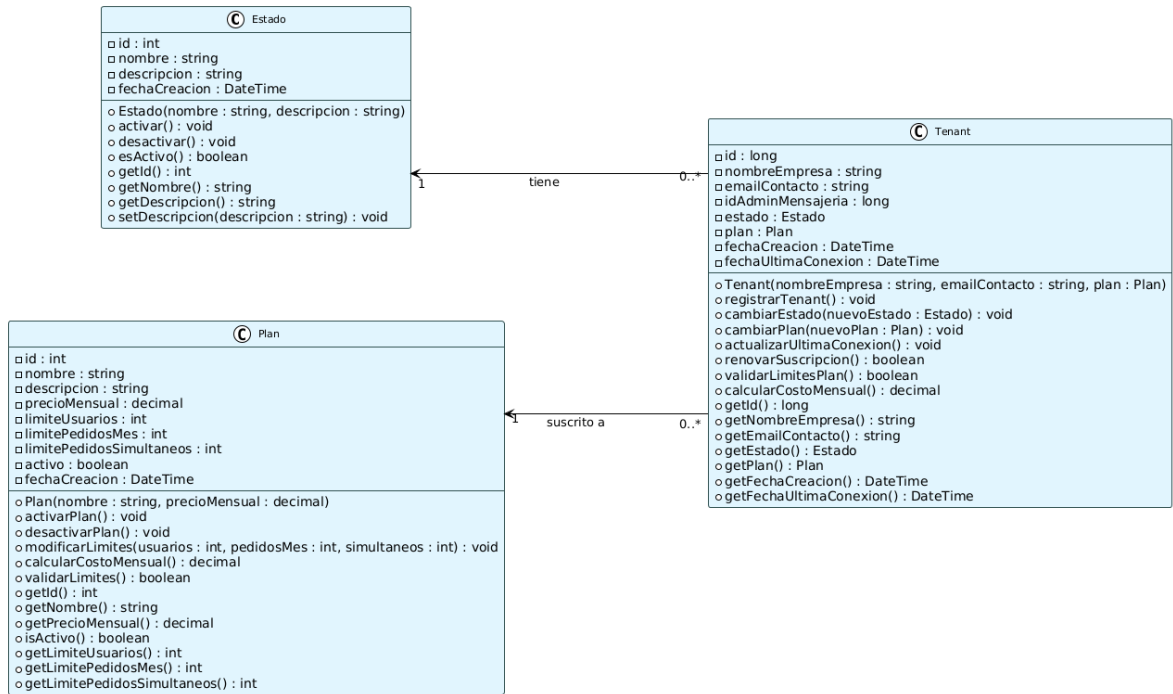


Figura 8

Diagrama de clases mensajería.

6.8 Diagrama de estados

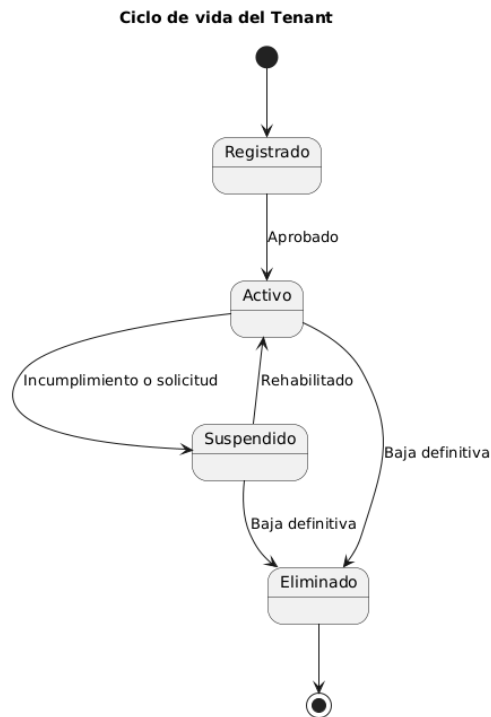
En esta sección se presentan los diagramas de estado correspondientes a dos grandes áreas del sistema: aquellos relacionados con la estructura multitenant y los correspondientes a las funciones básicas del funcionamiento del sistema.

6.8.1 Relacionados al tenant

1. Ciclo de vida del Tenant: Este diagrama ilustra el flujo del ciclo de vida de un tenant dentro del sistema, desde su creación hasta su desactivación. Se representa cómo el sistema gestiona cada fase, asegurando que el tenant pueda operar con todos los recursos necesarios.

Figura 9

Diagrama del ciclo de vida del tenant.

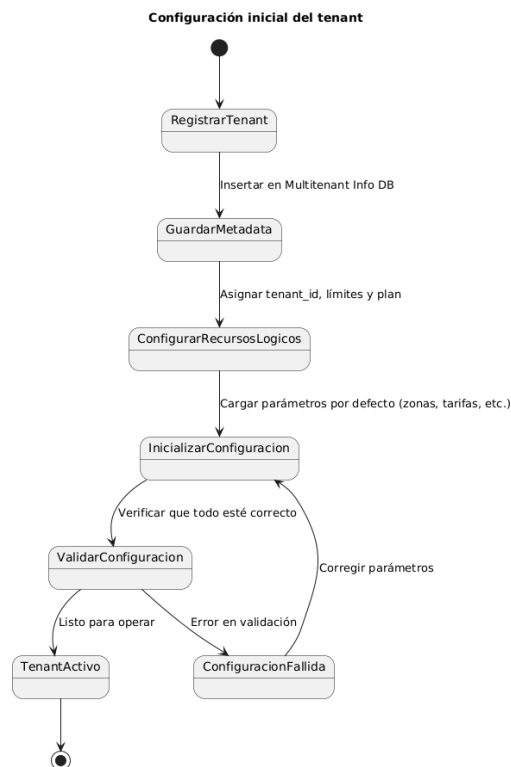


SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

2. Configuración inicial del tenant: Este diagrama muestra cómo se realiza la configuración inicial para un tenant en el sistema. Incluye la asignación de recursos, configuraciones básicas y la creación de parámetros de operatividad para que el tenant pueda comenzar a operar de forma óptima.

Figura 10

Diagrama configuración inicial del tenant.

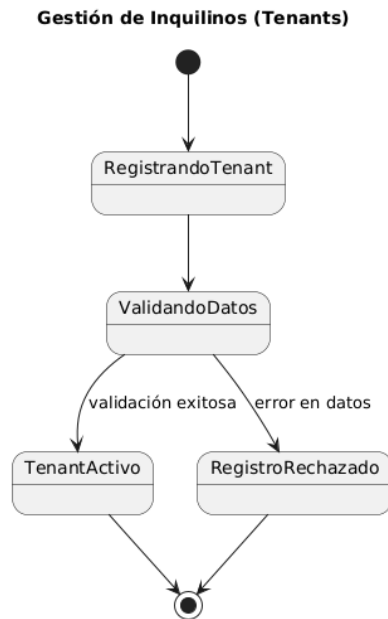


3. Gestión de registro de tenants: Este diagrama describe cómo se lleva a cabo la gestión y supervisión del registro de los diferentes tenants dentro del sistema.

Figura 11

Diagrama de gestión de inquilinos del tenant.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



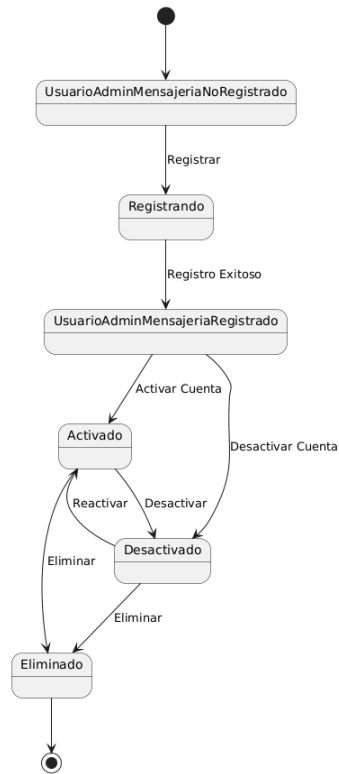
4. Ciclo de vida de un usuario del tenant: Representa el ciclo de vida de un usuario asociado a un tenant, desde su registro hasta su eliminación. El diagrama cubre las diferentes acciones del usuario y los estados a los que puede acceder dependiendo de su interacción con el sistema.

Figura 12

Diagrama de vida de un usuario del tenant.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Ciclo de vida de un administrador de mensajería

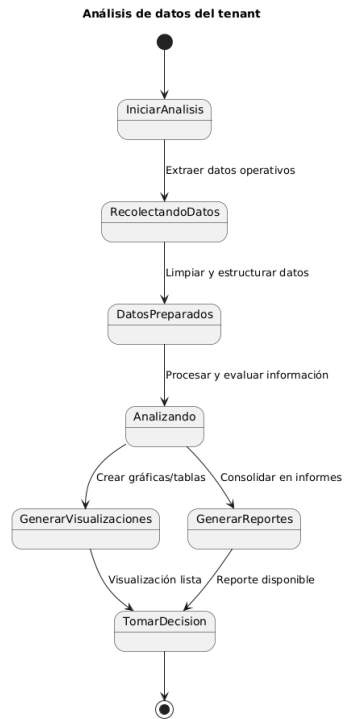


5. Análisis de datos del tenant: Aquí se muestra cómo se realiza el análisis y la visualización de datos de un tenant, incluyendo la recopilación de datos de uso, desempeño y otros aspectos operativos para la toma de decisiones estratégicas.

Figura 13

Diagrama de análisis de datos del tenant.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

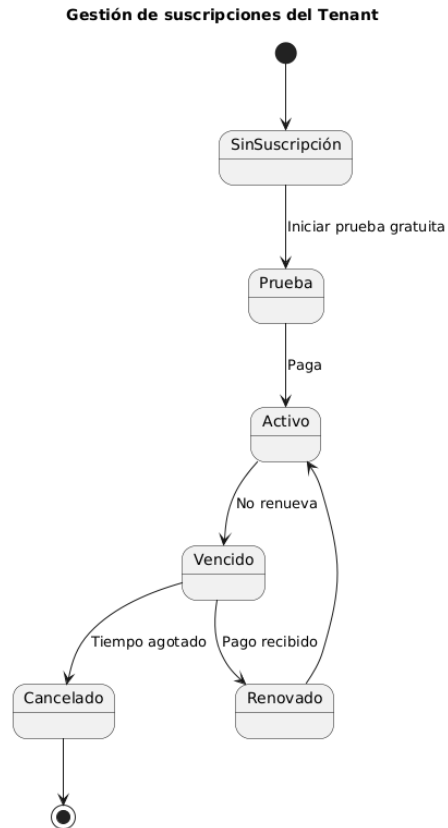


6. Gestión de suscripciones del tenant: Este diagrama refleja cómo se gestiona el proceso de suscripciones de un tenant dentro del sistema, incluyendo la creación, renovación y cancelación de suscripciones, así como los beneficios asociados a cada nivel.

Figura 14

Diagrama de gestión de suscripciones del tenant.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



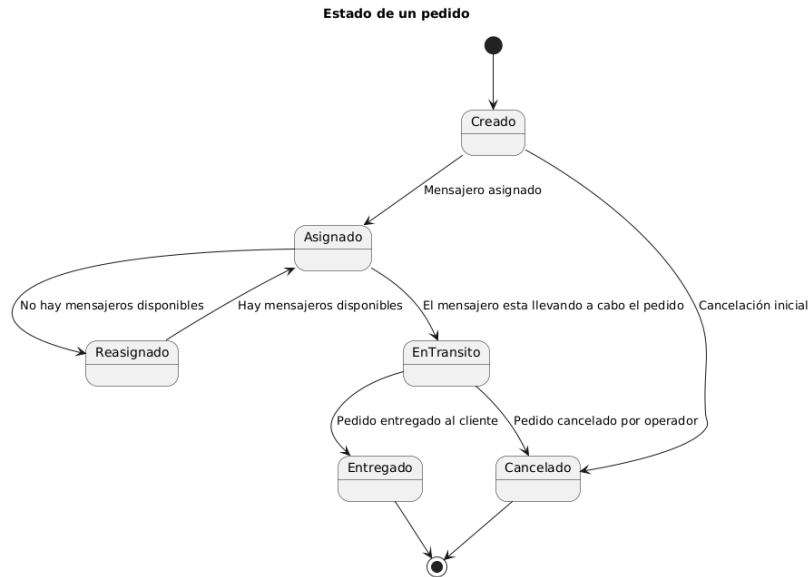
6.8.2 Funciones básicas del sistema

1. Estado pedido: Este diagrama describe los diferentes estados en los que un pedido puede encontrarse dentro del sistema, desde su creación hasta la entrega final.

Figura 15

Diagrama de estados de un pedido.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

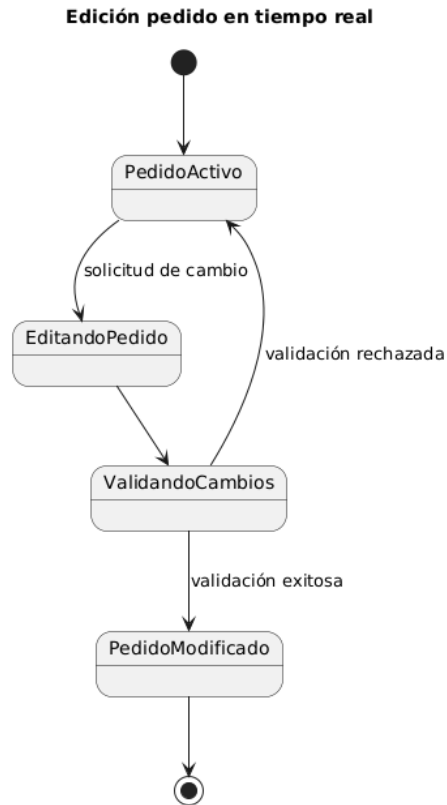


2. Edición pedido tiempo real: Este diagrama ilustra cómo se gestionan las modificaciones de un pedido en tiempo real, permitiendo que tanto los usuarios como los administradores pueden editar el pedido durante su proceso.

Figura 16

Diagrama de edición de un pedido en tiempo real.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

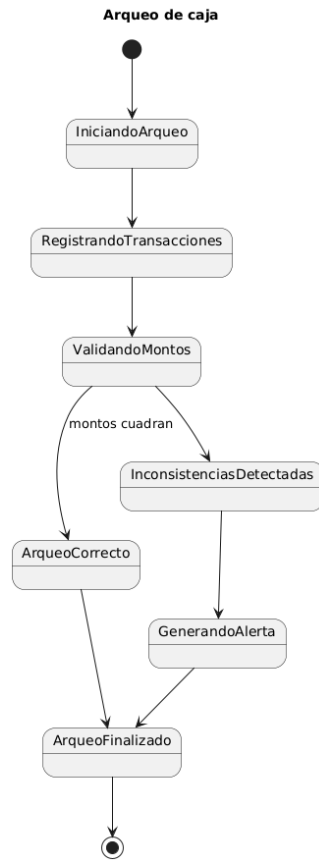


- Arqueo de caja: Este diagrama describe cómo se lleva a cabo el proceso de arqueo de caja dentro del sistema, permitiendo la verificación de los balances y la conciliación de los ingresos y gastos.

Figura 17

Diagrama de arqueo de caja.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



4. Gestión de roles y permisos: Aquí se ilustra cómo se gestionan los roles y permisos dentro del sistema, asegurando que cada usuario tenga acceso solo a las funcionalidades y datos que corresponden a su rol.

Figura 18

Diagrama de gestión de roles y permisos.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Gestión de roles y permisos

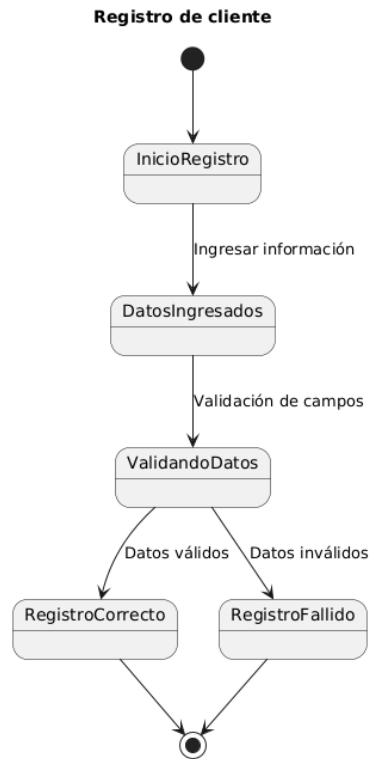


5. Registro de cliente: Este diagrama refleja el proceso de registro de un cliente, permitiendo a los usuarios disfrutar de beneficios exclusivos y un seguimiento detallado de sus interacciones con el sistema.

Figura 19

Diagrama de registro cliente.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

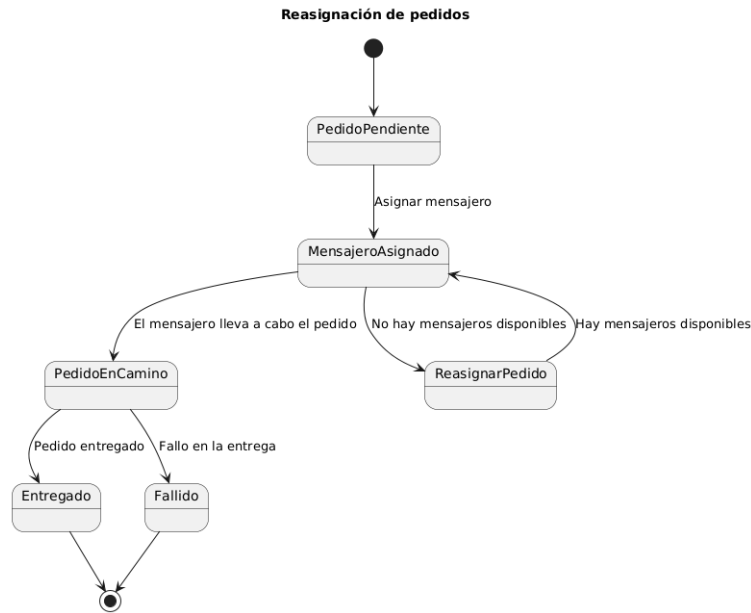


6. Reasignación de pedidos: Aquí se describe cómo un pedido puede ser reasignado a otro repartidor en caso de ser necesario, garantizando que el proceso de entrega continúe sin inconvenientes.

Figura 20

Diagrama de reasignación de pedidos.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

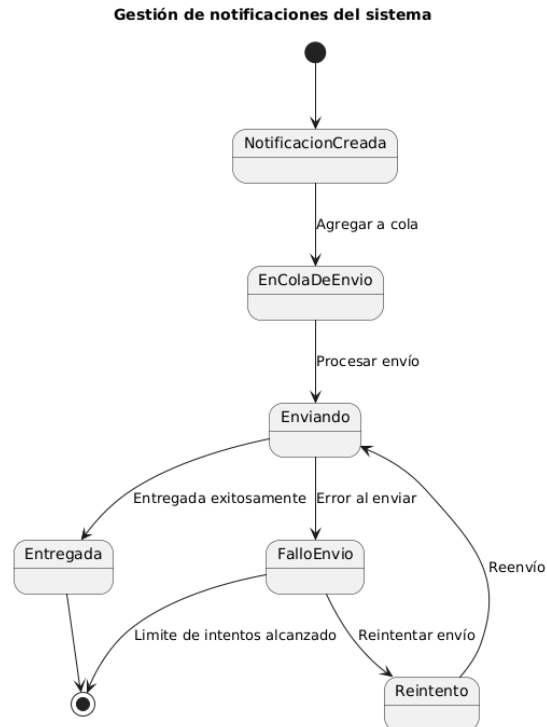


7. Gestión de notificaciones del sistema: Este diagrama refleja cómo el sistema gestiona las notificaciones, desde la creación de las notificaciones hasta su envío y confirmación de recepción.

Figura 21

Diagrama de gestión de notificaciones.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

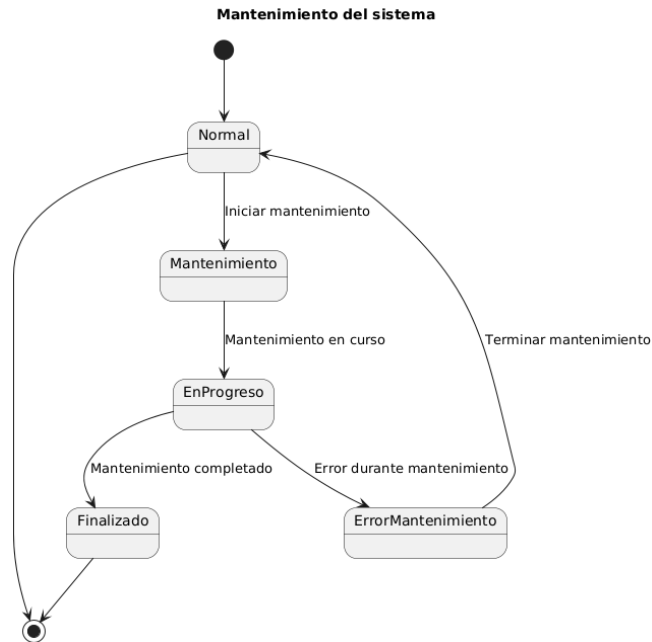


8. Mantenimiento del sistema: Finalmente, este diagrama describe el proceso de mantenimiento del sistema, abarcando desde la planificación del mantenimiento hasta la ejecución y la verificación de que el sistema vuelve a su estado operativo.

Figura 22

Diagrama del mantenimiento del sistema

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



6.9 Diagrama de actividades

Los diagramas de actividad que hemos desarrollado para tu aplicación de mensajería express muestran el flujo de los procesos clave dentro del sistema, ayudando a visualizar la interacción entre los usuarios y el sistema en diferentes escenarios operativos.

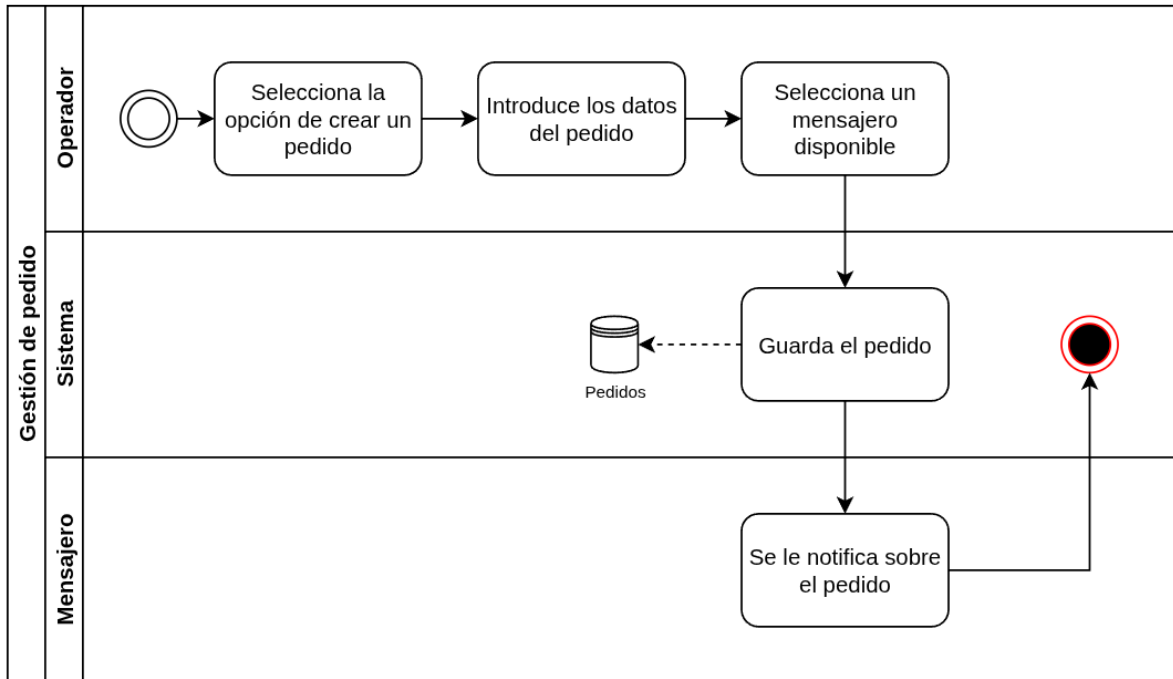
6.9.1 Creación de pedido

Representa el proceso mediante el cual un usuario genera un nuevo pedido en el sistema. Incluye la captura de información del cliente, la asignación de un mensajero y el envío de una notificación.

Importancia: Permite comprender cómo se gestionan los pedidos en tiempo real, asegurando eficiencia en la asignación de recursos.

Figura 23

Diagrama de actividad creación de pedidos



6.9.2 Edición de pedido en tiempo real

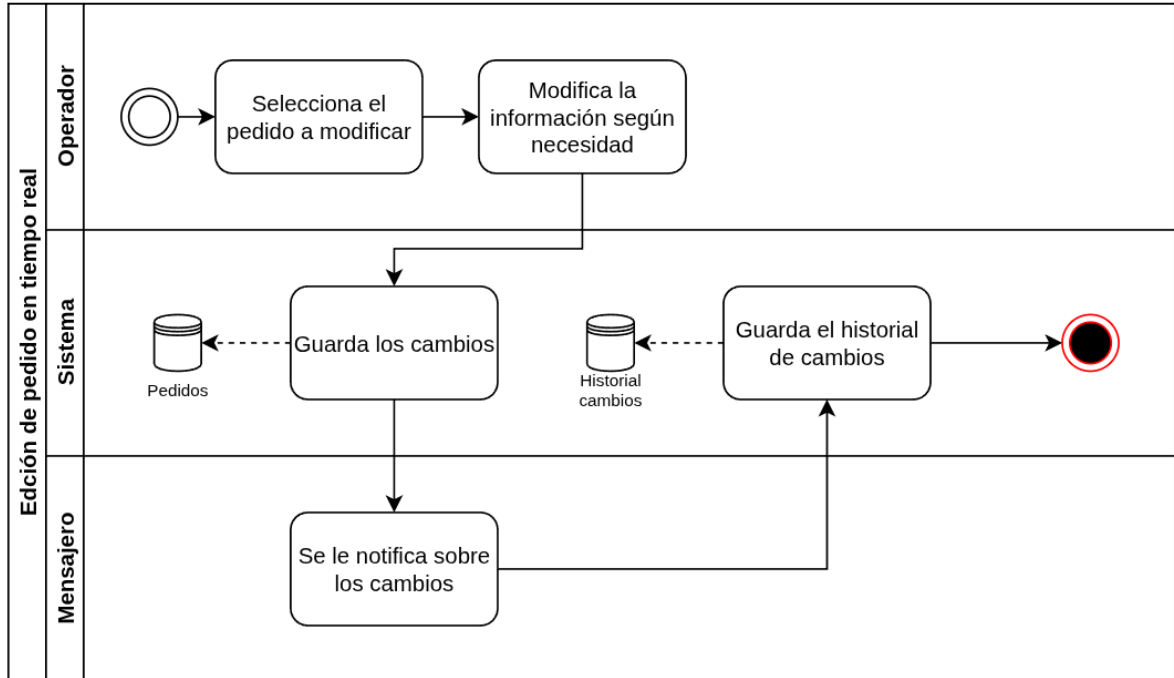
Muestra cómo un usuario puede modificar detalles de un pedido ya registrado. Puede incluir cambios en la dirección de entrega, asignación de un nuevo mensajero o actualización de costos.

Importancia: Garantiza flexibilidad en la gestión de pedidos y permite responder a cambios o imprevistos de los clientes.

Figura 24

Diagrama de actividad edición de pedidos en tiempo real

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



6.9.3 Confirmación de entrega de pedido

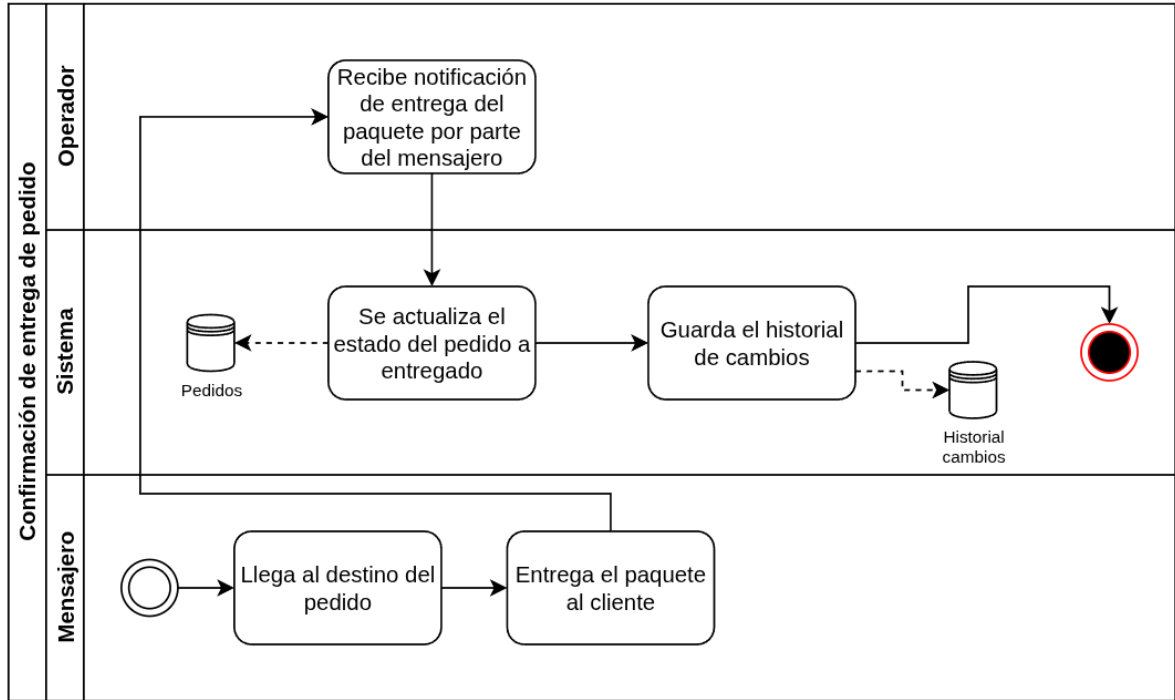
Representa el flujo de actividades desde que el mensajero llega a la dirección de entrega hasta que el pedido es marcado como entregado en el sistema. Incluye la actualización del estado del pedido.

Importancia: Es fundamental para el control de pedidos, asegurando que cada entrega se registre correctamente y se notifique a los involucrados.

Figura 25

Diagrama de actividad confirmación de entrega de pedido

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



6.9.4 Arqueo de caja

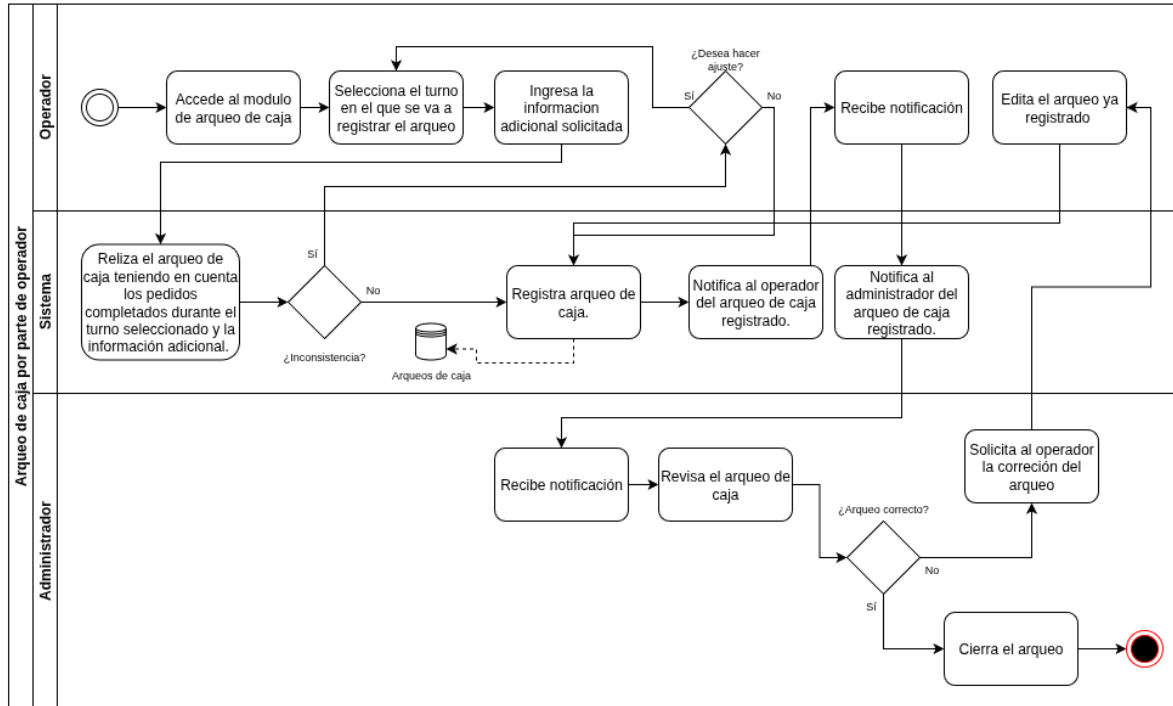
Modela el proceso mediante el cual un operador registra ingresos y egresos en el sistema, permitiendo generar un arqueo de caja detallado.

Importancia: Proporciona control financiero sobre las transacciones de mensajería, asegurando transparencia en los movimientos de dinero.

Figura 26

Diagrama de actividad arqueo de caja

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



6.9.5 Gestión de roles y permisos

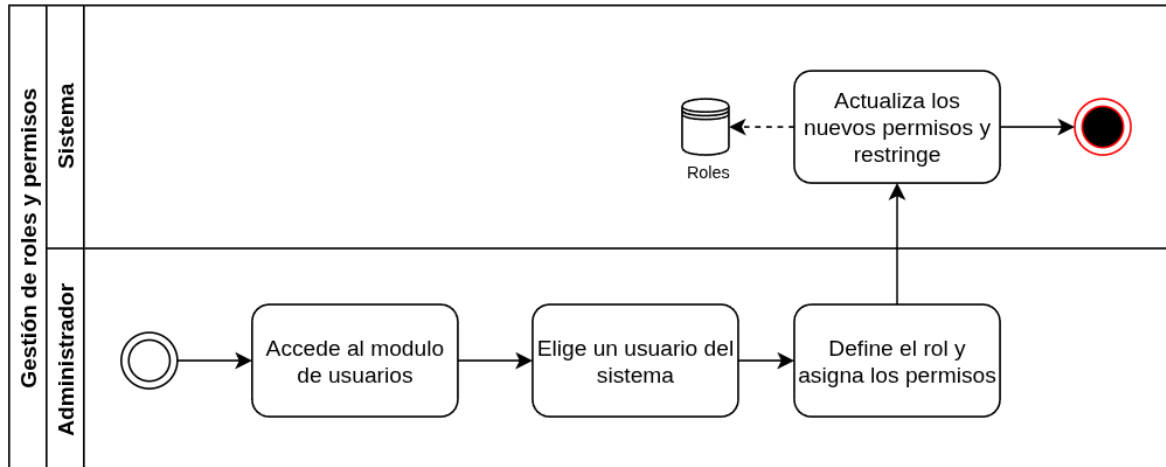
Representa cómo el administrador asigna y modifica los roles de los usuarios en el sistema. Incluye la actualización de permisos y restricciones de acceso a diferentes módulos según el rol.

Importancia: Asegura que cada usuario solo pueda acceder a las funcionalidades que le corresponden, mejorando la seguridad del sistema.

Figura 27

Diagrama de actividad gestión de roles y permisos

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



6.9.6 Registro de cliente

Muestra el proceso mediante el cual un operador registra nuevos clientes, almacena su información y les asigna beneficios como descuentos o tarifas preferenciales. A continuación, se modelan dos posibles situaciones.

Hay dos modalidades de registro de cliente: usando datos ya existentes (de un pedido registrado en el sistema) o con nuevos datos.

Importancia: Facilita la gestión de clientes recurrentes y mejora la personalización del servicio.

Figura 28

Diagrama de actividad registro del cliente (datos existentes)

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

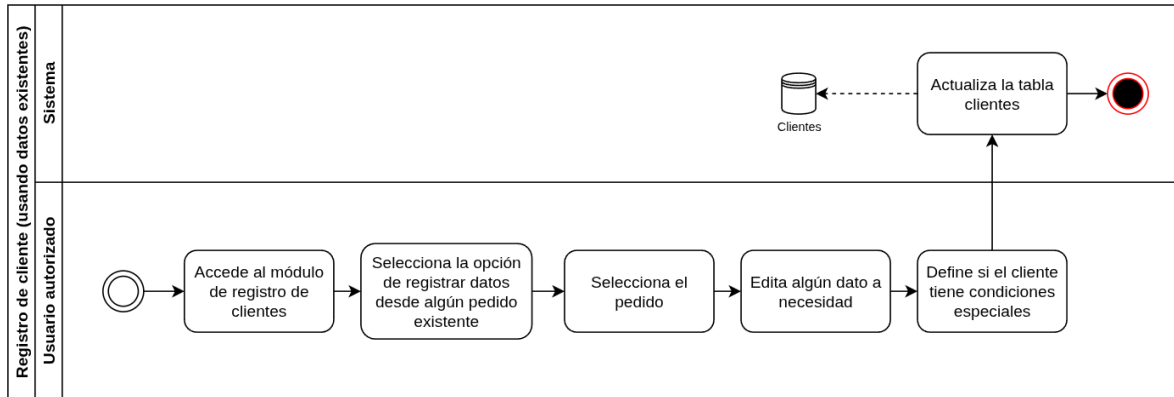
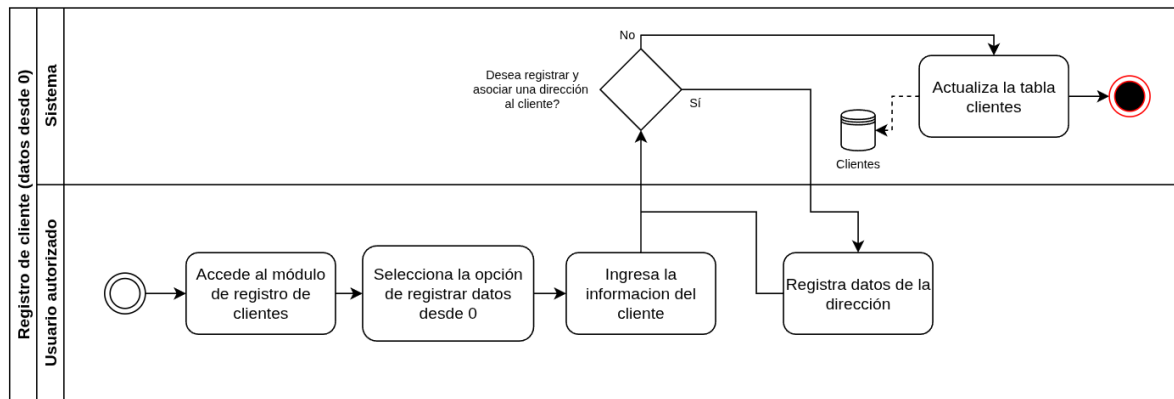


Figura 29

Diagrama de actividad registro del cliente (nuevos datos)



7. Diseño de aplicación

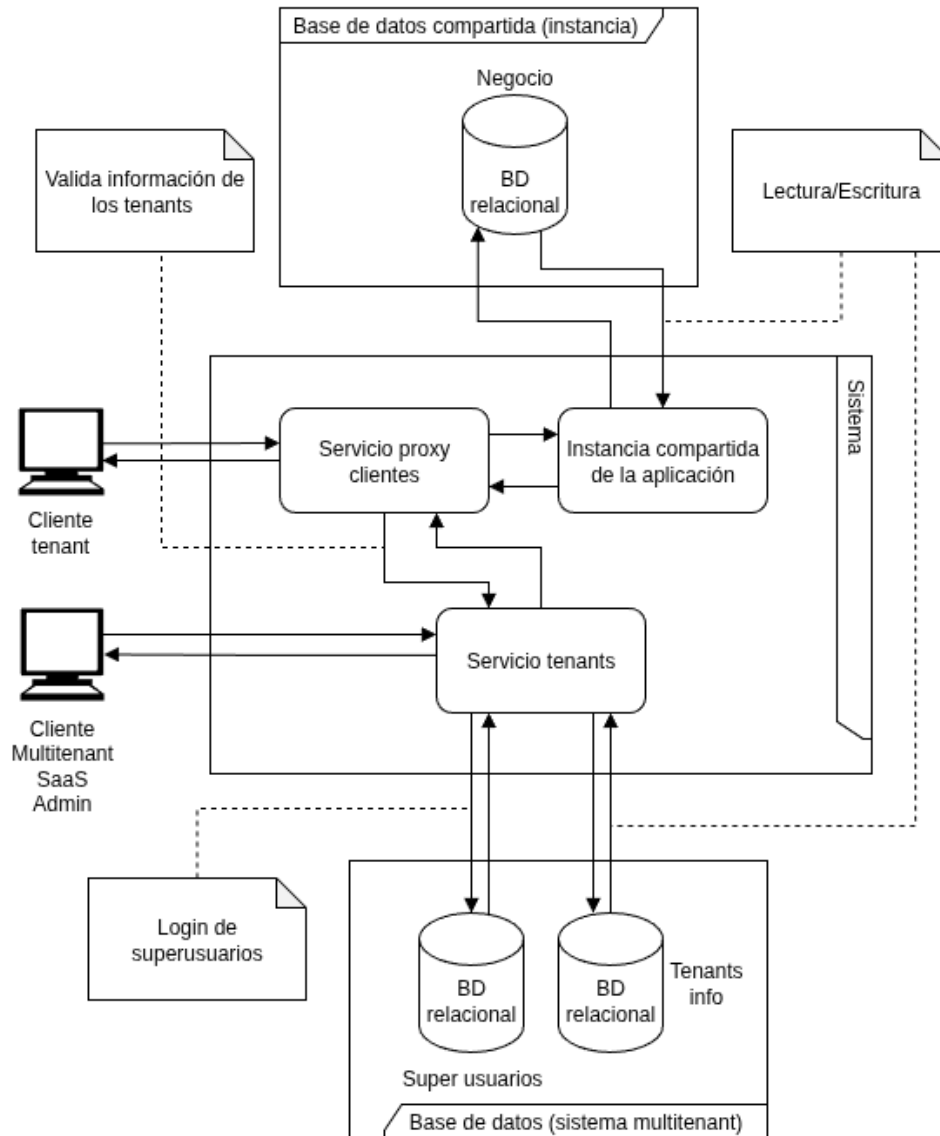
7.1 Arquitectura planteada

Tal como se mencionó en apartados anteriores, para implementar el prototipo de software usando una arquitectura multitenant, se escogió el enfoque de instancia y base de datos compartida, así pues, se presenta a continuación la arquitectura final planteada.

Figura 30

Diagrama de arquitectura final

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



Se puede identificar los siguientes componentes:

- Cliente (frontend) de los inquilinos (tenant).
- Cliente (frontend) de los administradores del sistema multitenant.
- Instancia de base de datos del sistema multitenant: La cual se subdivide en dos bases de datos, una donde se aloja la información de los usuarios (super

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

administradores) y una segunda en la que se almacena la información de los tenants (planes, administrador del tenant, activo/inactivo, entre otros).

- Instancia de base de datos compartida que almacena todos los datos relacionados con los procesos de negocio de cada inquilino. Al adoptar este enfoque de base de datos compartida, es crítico que cada registro incluya un campo identificador (tenant_id) que permita distinguir los datos de un inquilino de los demás.
- Servicio (Backend) de tenants el cual facilita la comunicación entre el cliente de los administradores y la instancia de la base de datos del sistema multitenant. A su vez, valida los datos de los tenants asociados a las peticiones que han sido redirigidas por parte del servicio proxy clientes.
- Servicio (proxy) clientes. Se define como el servicio más importante de la arquitectura. Funciona como mediador entre el cliente de los inquilinos y el servicio de la aplicación compartida ya que tiene la capacidad de interceptar la petición por parte del cliente, redirigir esta petición al servicio proxy para hacer las validaciones necesarias referente a la información de los tenants y en base a la respuesta que se obtenga dejar pasar la petición al servicio de la aplicación compartida o en su defecto denegar la petición de ser necesario. Podría cumplir funciones adicionales como por ejemplo servir de balanceador de cargas o capturar datos de gran relevancia para los administradores del sistema multitenant como lo puede ser datos de uso de red y tráfico de peticiones (asociadas a cada tenant) en el sistema.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Servicio backend (de la instancia compartida) el cual implementa la funcionalidad central del sistema y aplica las reglas de negocio, procesando las peticiones filtradas por el servicio proxy desde los clientes de los tenants y gestionando su comunicación con la base de datos compartida.

En el prototipo implementado, cada componente se ejecuta en un contenedor Docker independiente, interconectados mediante una red interna que gestiona Docker. La comunicación entre ellos se realiza mediante direcciones IP asignadas automáticamente bajo la convención de nomenclatura propia de la herramienta, siendo así:

- Cliente de los inquilinos `http://react-frontend-courier:80`
- Cliente de los administradores del sistema multitenant `http://react-frontend-tenants:80`
- Instancia de base de datos del sistema multitenant `http://mysql-tenant:3306`
- Instancia de base de datos compartida `http://mysql-mensajeria:3306`
- Servicio de tenants `http://tenant-app:8080`
- Servicio proxy clientes `http://middleware-service:8082`
- Servicio de la instancia compartida de la aplicación `http://courier-app:8081`

7.2 Backend

La capa del backend constituye la parte fundamental del software que permite el acceso y manipulación de los datos gestionados por la aplicación de mensajería express. Para el caso del sistema de mensajería se estableció que se iba a implementar una arquitectura basada en microservicios desarrollados en Spring Boot, donde cada microservicio se encarga de manejar

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

las diferentes peticiones que llegan a través de un RESTful API específico de su dominio. Esto implica que las peticiones que hace el cliente llegan mediante endpoints o rutas definidas por cada aplicación, para responder a estas peticiones se hace uso de una representación del estado del recurso al cual se está queriendo acceder. Cada API se encarga de retornar esta representación a la aplicación del lado del cliente por medio de protocolo HTTP y por lo general el recurso se representa en formato JSON.

Se decide hacer la implementación de APIs RESTful ya que son interfaces de programación que se adaptan de forma óptima a la arquitectura de microservicios establecida, permitiendo la comunicación efectiva entre las aplicaciones por parte del cliente y los servidores Web especializados. El manejo de la información que ocurre al interior de una arquitectura REST permite que exista uniformidad en los diferentes recursos de la aplicación y en sus representaciones, sumado a una gran flexibilidad para la visualización de información por parte del cliente para consultas o para la manipulación de la misma.

7.2.1 Arquitectura de microservicios

El sistema backend está estructurado mediante tres microservicios especializados que trabajan en conjunto para proporcionar funcionalidad completa al sistema multitenant. Esta separación de responsabilidades permite un mantenimiento más eficiente, escalabilidad independiente y una clara delimitación de dominios de negocio.

7.2.1.1 Microservicio de tenants. El microservicio de tenants constituye el núcleo de control del sistema multitenant, encargándose exclusivamente de la gestión de inquilinos

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

(empresas de mensajería) y planes de suscripción. Este servicio mantiene la información crítica que determina qué tenants pueden acceder al sistema y bajo qué condiciones operativas.

7.2.1.2 Microservicio de mensajería. El microservicio de mensajería implementa toda la lógica de negocio específica del dominio de mensajería express. Se encarga de gestionar pedidos, clientes, mensajeros, tarifas, arqueos de caja y todas las operaciones operativas que realizan las empresas de mensajería en su funcionamiento diario.

7.2.1.3 Servicio proxy de cliente. El servicio proxy está desarrollado en Node.js con Express.js y actúa como intermediario crítico entre los frontends de inquilinos y el microservicio de mensajería. Su función principal es interceptar las peticiones, validar el estado de los tenants mediante comunicación HTTP con el microservicio de tenants y controlar el acceso a los recursos según las políticas establecidas para cada empresa de mensajería. Implementa lógica diferenciada para usuarios superadministradores.

7.2.2 Protocolo de comunicación

Para la comunicación entre las interfaces de usuario hasta los servidores de las aplicaciones se utiliza el protocolo HTTP (Protocolo de Transferencia de Hipertexto). HTTP se adapta al modelo propuesto para la solución de mensajería express multitenant, permite a los usuarios realizar peticiones y esperar a que los servidores correspondientes retornen una respuesta. El protocolo HTTP facilita la implementación del patrón request-response necesario para las operaciones de gestión de pedidos, seguimiento de entregas, administración de recursos del sistema de mensajería y gestión de tenants.

7.2.3 Tipos de peticiones HTTP soportadas

Las peticiones que soportan los servidores son de 5 tipos principales (GET, POST, PUT, PATCH y DELETE), con estos cinco tipos de peticiones es posible dar cumplimiento a los requerimientos funcionales definidos para la aplicación de mensajería express multitenant:

- Petición **GET**: Le permite al usuario obtener la información de tenants, empresas de mensajería, usuarios, clientes, pedidos, mensajeros, tarifas, arqueos e historial de pedidos por medio de parámetros de búsqueda. Esta petición es fundamental para consultar el estado actual de los pedidos, verificar la disponibilidad de mensajeros, obtener reportes de gestión y validar el estado de los tenants.
- Petición **POST**: Permite al usuario crear nuevos registros de tenants, empresas, usuarios, clientes, pedidos, mensajeros y tarifas, persistiéndolos en el servidor correspondiente a partir de la información ingresada por los administradores u operadores.
- Petición **PUT**: Permite modificar completamente la información existente de tenants, empresas, usuarios, clientes, pedidos, mensajeros y tarifas a partir de la información actualizada ingresada por el usuario, persistiendo los cambios en el servidor correspondiente. Es especialmente importante para actualizar estados de pedidos, cambiar asignaciones de mensajeros y modificar información de contacto.
- Petición **PATCH**: Permite modificaciones parciales de recursos, especialmente útil para actualizaciones específicas como cambios de estado de pedidos,

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

actualización de disponibilidad de mensajeros, modificación de campos específicos de clientes o ajustes menores en configuraciones de tenants sin afectar otros campos del recurso.

- Petición **DELETE**: Permite eliminar la información asociada a tenants, empresas, usuarios, clientes, pedidos, mensajeros y tarifas en el servidor correspondiente, o deshacer relaciones existentes entre dos recursos. Esta funcionalidad está restringida según el rol del usuario.

7.2.4 Manejo de parámetros y autenticación

El uso del protocolo HTTP permite el aprovechamiento de parámetros y cabeceras para las diferentes peticiones que llegan a los servidores. A través de los parámetros de las peticiones resulta posible realizar filtros al momento de buscar recursos en las aplicaciones u obtener un recurso en particular haciendo uso de su identificador único por medio de la ejecución de queries en las bases de datos correspondientes. Por ejemplo, es posible filtrar pedidos por estado o validar el estado de un tenant particular.

Por otra parte, a través de las cabeceras de las peticiones es posible validar la autenticación de los usuarios por medio de tokens de autorización diferenciados según el microservicio. Con cada petición que realice el usuario se adjunta el token que le fue asignado al momento de iniciar sesión en la aplicación al interior de la cabecera de la petición. Sin el uso de este token el usuario no está autorizado para realizar operaciones a través de los servidores web. Este mecanismo es crucial para mantener la seguridad en un entorno multitenant donde cada

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

empresa de mensajería debe acceder únicamente a su propia información y donde los superadministradores requieren tokens especializados para gestionar tenants.

El servicio proxy implementa validación adicional mediante el header x-additional-data que contiene información del tenant y contexto del usuario, permitiendo validaciones específicas antes de reenviar las peticiones al microservicio de mensajería.

7.2.5 Arquitectura basada en componentes

Para el diseño de los microservicios de Spring Boot se utiliza una estructura basada en componentes siguiendo el patrón de arquitectura en capas. El servicio proxy, desarrollado en Node.js, implementa una arquitectura diferente basada en middleware de Express.js. Cada componente se encarga de una funcionalidad específica para la aplicación y se comunican entre ellos para resolver cada petición que recibe el servidor correspondiente.

7.2.5.1 Capa de controladores y rutas. Los controllers de cada microservicio se encargan de definir las rutas en las que van a estar expuestos los endpoints del API REST correspondiente. Definen qué parámetros va a recibir la petición y si van a estar protegidos bajo algún tipo de autenticación específica del dominio.

Controladores del microservicio de tenants:

- TenantController (/api/tenants/): Maneja las operaciones CRUD de tenants.
- PlanController (/api/planes/): Gestiona los planes de suscripción.
- EstadoController (/api/estados/): Se encarga de los estados del tenant.
- ValidationController (/api/validation/): Validación de tenants.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Controladores principales del microservicio de mensajería:

- AuthController (/api/auth/): Maneja la autenticación de usuarios operativos.
- ClienteController (/api/clientes/): Gestiona las operaciones CRUD para clientes.
- EmpresaMensajeriaController (/api/empresas-mensajeria/): Administra las empresas de mensajería.
- MensajeroController (/api/mensajeros/): Controla la gestión de mensajeros.
- PedidoController (/api/pedidos/): Maneja las operaciones de pedidos.
- TarifaController (/api/tarifas/): Gestiona las tarifas de servicio.
- ArqueoCajaController (/api/arqueo-caja/): Control de arqueos de caja.
- HistorialPedidoController (/api/historial-pedidos/): Acceso al historial de cambios de los pedidos.
- UsuarioController (/api/usuarios/): Gestiona la administración de usuarios.
- DashboardGeneralController (/api/dashboard/): Proporciona métricas y estadísticas.
- NotificacionController (/api/notificaciones/): Manejo de notificaciones.

Rutas del servicio proxy (Node.js):

- server.js: Versión básica con validación estándar de tenants.
- server-maintain.js: Versión avanzada con manejo diferenciado para superadministradores y soporte para archivos binarios.

Rutas principales:

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Ruta universal (app.all('*')): Intercepta todas las peticiones y actúa como proxy middleware.
- /health: Endpoint de monitoreo del estado del servicio.
- Proxy transparente: Reenvía todas las rutas al microservicio de mensajería tras validación.

Algunos de los endpoints de las aplicaciones pueden recibir identificadores de los registros en las bases de datos, representaciones de un recurso en formato JSON o requerir algún token de autenticación del usuario específico según su rol y el microservicio correspondiente.

7.2.5.2 Capa de servicios y funciones. Cuando una petición se hace de forma adecuada al endpoint definido en el controller, este realiza un llamado al service correspondiente, los services de cada microservicio se encargan de definir la respuesta que se le envía al usuario que realiza la petición, ya sea un mensaje de error o información proveniente de la base de datos correspondiente.

Servicios del microservicio de tenants: Los services se encargan de ejecutar operaciones específicas del dominio multitenant, tales como:

- Validación de límites de planes de suscripción.
- Verificación de estado activo/inactivo de tenants.
- Gestión de configuraciones específicas por tenant.
- Sincronización de información con otros microservicios.

Servicios del microservicio de mensajería: Los services ejecutan las operaciones del negocio específicas del dominio de mensajería, tales como:

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Lógica de asignación inteligente de pedidos a mensajeros.
- Validación de reglas de negocio para transiciones de estados.
- Gestión automática de disponibilidad de mensajeros.
- Cálculo de métricas de rendimiento y eficiencia operativa.
- Generación de reportes financieros y operativos.
- Aplicación de tarifas dinámicas y descuentos.

Funciones del servicio proxy (Node.js): El servicio proxy implementa funciones especializadas en lugar de services tradicionales:

- `validateWithTenant(additionalData, authToken)`: Función que realiza validación HTTP con el microservicio de tenants, verifica el estado 'ACTIVO' del tenant mediante petición GET a `/api/tenants/{tenantId}/info`.
- `extractAdditionalData(req)`: Extrae y parsea información del header `x-additional-data` que contiene: `mensajeríaId`, `tenantId`, `nombreUsuario`, `rol`.
- `decodeJwt(token)`: Decodifica tokens JWT para extraer información de roles.
- `makeRequestToCourier(config, isBinary)`: Función para realizar peticiones HTTP al microservicio de mensajería con soporte para respuestas JSON y binarias.
- `isBinaryResponse(headers)`: Detecta automáticamente respuestas binarias (PDF o Excel) basándose en `Content-Type`.

7.2.5.3 Capa de repositorios y clientes HTTP. La capa de repositorios utiliza Spring Data JPA para proporcionar una abstracción de acceso a datos especializada según el dominio de cada microservicio.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Repositorios del microservicio de tenants: Cada entidad del dominio multitenant (Tenant, Plan, Estado) tiene su repositorio correspondiente que extiende de JpaRepository, proporcionando operaciones CRUD básicas y consultas personalizadas para validaciones de tenants.

Repositorios del microservicio de mensajería: Cada entidad del dominio de mensajería (Empresa, Usuario, Cliente, Pedido, Mensajero, Tarifa, ArqueoCaja, HistorialPedido, Notificacion) tiene su repositorio correspondiente con consultas especializadas para operaciones complejas como estadísticas de rendimiento, reportes financieros y consultas multitenant con filtrado automático por tenant_id.

Cliente HTTP del servicio proxy (Node.js): El servicio proxy utiliza Axios como cliente HTTP para comunicación entre microservicios:

- Configuración base: Timeout de 10 segundos para validaciones con tenant-service, 30 segundos para operaciones con courier-service.
- Headers management: Limpieza automática de headers internos ('x-additional-data', 'host') antes del reenvío.
- Error handling: Gestión especializada de códigos de error HTTP y errores de conectividad.
- Response streaming: Soporte para respuestas binarias mediante streaming directo para mantener integridad de archivos.
- Configuración de URLs: Variables de entorno para endpoints de tenant-service (puerto 8080) y courier-service (puerto 8081).

7.2.5.4 Capas adicionales de la arquitectura.

Capa de DTOs (Data Transfer Objects): Cada microservicio define objetos de transferencia de datos específicos para su dominio, garantizando que solo la información necesaria se transfiera entre capas y servicios, mejorando la seguridad y el rendimiento.

Capa de mappers: Utiliza herramientas como MapStruct para transformar automáticamente entre entidades de base de datos y DTOs, manteniendo la separación de responsabilidades y facilitando el mantenimiento.

Capa de request/response: Define estructuras específicas para peticiones y respuestas de cada endpoint, proporcionando validaciones automáticas y documentación clara de las APIs.

Capa de configuración: Gestiona configuraciones específicas de cada microservicio.

Capa de seguridad: Implementa mecanismos de autenticación y autorización específicos para cada dominio, con tokens JWT diferenciados y validaciones de permisos granulares.

7.2.6 Gestión de base de datos

El sistema utiliza dos bases de datos especializadas que soportan la arquitectura multitenant, donde cada microservicio accede a la información específica de su dominio, se validan las reglas de negocio correspondientes y la integridad de los datos ingresados por el usuario, se obtiene la respuesta a la petición del cliente y en caso de existir cambios en los

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

registros o la creación de un nuevo registro, se hace la persistencia de la información en la base de datos correspondiente.

El servicio proxy no mantiene persistencia propia, sino que actúa como intermediario HTTP entre los clientes y los microservicios que sí gestionan bases de datos.

7.2.6.1 Base de datos de tenants. Esta base de datos especializada almacena exclusivamente información relacionada con la gestión multitenant, incluyendo datos de inquilinos, planes de suscripción, estados de tenants y configuraciones del sistema. El modelo implementa las siguientes tablas principales.

- tenant: Define cada inquilino del sistema con información empresarial y referencias a planes.
- plan: Gestiona planes de suscripción con límites configurables.
- estado: Control de estados para tenants.

7.2.6.2 Base de datos de mensajería. El modelo de base de datos relacional manejado por el microservicio de mensajería fue diseñado para soportar un entorno multitenant mediante base de datos compartida, donde cada empresa de mensajería mantiene sus datos completamente aislados a través del campo tenant_id. El esquema implementa las siguientes tablas principales.

- empresa_mensajería: Tabla maestra que define cada empresa dentro de un tenant.
- usuario: Almacena información de usuarios con roles diferenciados (admin_mensajería, operador, mensajero).

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- mensajero: Tabla especializada que extiende la información de usuarios con rol mensajero.
- cliente: Contiene la información de clientes asociados a cada empresa de mensajería.
- pedido: Gestiona los pedidos con su ciclo de vida completo y trazabilidad.
- historial_pedido: Mantiene un registro de auditoría de todos los cambios realizados a un pedido.
- tarifa: Define estructuras de precios por empresa de mensajería.
- arqueo_caja: Control financiero con seguimiento de ingresos y egresos.

7.2.7 Integridad y constraints

En los scripts de creación de las tablas de ambas bases de datos se establecen los nombres y tipos correspondientes para cada columna junto con las restricciones existentes de llaves primarias, llaves foráneas, columnas con valores nulos y las validaciones de reglas de integridad de los datos por medio de restricciones de tipo check y unique. Adicionalmente, se implementan triggers automáticos que garantizan:

- Sincronización automática: Creación de registros en la tabla mensajero cuando se asigna el rol correspondiente.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Gestión de disponibilidad: Actualización automática del estado de disponibilidad según asignaciones de pedidos.
- Auditoría completa: Registro automático de cambios en el historial para trazabilidad.

7.2.8 Aislamiento multitenant

El diseño implementa aislamiento a nivel de fila (Row-Level Security) donde cada registro incluye una referencia a la empresa de mensajería correspondiente (`mensajeria_id`), garantizando que los usuarios solo puedan acceder a los datos de su organización. Esta estrategia proporciona un balance óptimo entre aislamiento de datos, eficiencia de recursos y facilidad de mantenimiento.

7.2.9 Manejo de transacciones y consistencia

El sistema utiliza las capacidades transaccionales de Spring Boot en cada microservicio para garantizar la consistencia de los datos dentro de su dominio específico. El servicio proxy al ser stateless y no manejar persistencia, no requiere manejo transaccional directo, pero garantiza la integridad de las comunicaciones HTTP mediante timeouts, retry policies y manejo robusto de errores.

Para operaciones que involucran múltiples microservicios, se implementan patrones de consistencia eventual y mecanismos de compensación:

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Transacciones locales, cada microservicio mantiene consistencia transaccional en operaciones críticas como:
 - Creación y modificación de tenants con validación de límites.
 - Asignación de pedidos a mensajeros con actualización de disponibilidad.
 - Cambios de estado de pedidos con registro en historial.
 - Actualización de arqueos de caja con cálculo automático de totales.
- Consistencia entre microservicios, para operaciones distribuidas se implementan:
 - Validaciones síncronas críticas dado que el proxy realiza validaciones HTTP síncronas con el microservicio de tenants antes de permitir operaciones.
 - Eventos de notificación para sincronización de cache y actualizaciones de estado.
 - Mecanismos de retry configurados en el proxy mediante timeouts y manejo de errores ECONNREFUSED/ECONNABORTED.
 - Circuit breaker pattern implementado mediante timeout y validateStatus en configuración de Axios.
- Garantías del servicio proxy:
 - La validación con tenant-service es atómica, se valida completamente o falla.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Hay integridad de transmisión, especialmente para archivos binarios mediante streaming directo.
- Cuenta con aislamiento de tenants, solo usuarios validados pueden acceder a recursos de su tenant específico.
- Logging de errores y validaciones para auditoría y debugging.

7.3 Frontend

La capa del frontend constituye la interfaz de usuario del sistema de mensajería express, ejecutándose del lado del cliente y proporcionando una experiencia interactiva y funcional para los diferentes tipos de usuarios. Esta capa es responsable de la visualización de la información que maneja la aplicación, el ingreso de datos por parte de los usuarios y la comunicación con el backend a través de servicios REST.

Para el desarrollo del sistema de mensajería se implementaron dos aplicaciones frontend independientes pero complementarias, cada una diseñada para atender necesidades específicas del sistema multi-tenant. Ambas aplicaciones son aplicaciones web modernas desarrolladas para ser ejecutadas en navegadores web, las cuales a través de conexión a internet permiten a los usuarios interactuar con los componentes gráficos de la interfaz de usuario para accionar operaciones en el servidor web mediante el acceso a los endpoints definidos en el RESTful API.

7.3.1 Arquitectura dual del frontend

7.3.1.1 Frontend de gestión de tenants. El primer frontend está dedicado exclusivamente a la gestión de tenants (empresas de mensajería) y constituye el nivel más alto de administración del sistema. Esta aplicación está diseñada para ser utilizada únicamente por superadministradores, quienes tienen la responsabilidad de supervisar y gestionar el funcionamiento global del sistema multi-tenant.

7.3.1.2 Funcionalidades principales.

- **Gestión de tenants:** Permite crear, modificar, activar, desactivar y eliminar empresas de mensajería (tenants). Los superadministradores pueden configurar los parámetros iniciales de cada tenant, establecer límites de uso y monitorear el estado general de cada empresa registrada en el sistema.
- **Administración de administradores:** Proporciona herramientas para gestionar los usuarios administradores asignados a cada tenant, incluyendo la creación de cuentas, desactivación de estas e incluso eliminación de perfiles.
- **Gestión de perfil:** Permite a los superadministradores modificar su información personal y configuraciones de cuenta.

7.3.2 Arquitectura y tecnologías

Para el diseño de ambos frontends se utilizó una estructura basada en componentes de React, aprovechando las ventajas de Vite como herramienta de construcción para mejorar la

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

velocidad de desarrollo y el rendimiento de las aplicaciones. El diseño está implementado de tal forma que los usuarios puedan navegar entre los diferentes módulos de forma intuitiva y con fluidez.

Las aplicaciones utilizan React Router para el manejo de rutas y navegación protegida, Bootstrap para el diseño responsivo y Axios para el consumo de servicios del API REST. Esta arquitectura garantiza un rendimiento óptimo y una experiencia de usuario consistente en ambas aplicaciones.

Cada componente se encarga de permitirle al usuario ejecutar cada una de las funcionalidades de la aplicación para dar cumplimiento a los requerimientos funcionales definidos para el sistema de mensajería. Esto es posible ya que la interacción con cada uno de los componentes de la interfaz genera el envío de peticiones HTTP a los endpoints del API REST a través del proxy para que la petición sea ejecutada en el servidor Web.

7.3.3 Sistema de roles y permisos

7.3.3.1 Rol de administrador. Los administradores tienen acceso completo a todas las funcionalidades operativas del sistema de mensajería dentro de su tenant. Sus módulos disponibles incluyen:

- Arqueos de caja: Gestión completa de los arqueos de caja con capacidad para generar reportes detallados de ingresos, y egresos, además de que son los únicos usuarios con la capacidad de cerrar un arqueo tras revisión.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Dashboard general: Panel de control con estadísticas comprehensivas del negocio, incluyendo métricas de pedidos, rendimiento de mensajeros e ingresos, los cuales se muestran con gráficos y pueden consultarse con fechas personalizadas.
- Gestión de clientes: Módulo completo para administrar la base de datos de clientes, incluyendo creación, edición y eliminación de estos, se tiene en cuenta la opción de darles descuentos y de crearse en base a pedidos sin cliente.
- Gestión de mensajeros: Administración del personal de mensajería, incluyendo edición de su vehículo de trabajo y su disponibilidad tanto por pedidos como por cuestiones personales.
- Gestión de pedidos: Control total sobre el ciclo de vida de los pedidos, desde su creación hasta la entrega final, junto a la funcionalidad de historial para ver cuáles fueron los cambios y quienes los realizó, también tiene implementada una funcionalidad para ver la ruta del pedido.
- Gestión de tarifas: Configuración y mantenimiento del sistema de precios y tarifas para diferentes tipos de servicios junto a su activación o desactivación.
- Gestión de usuarios: Administración de cuentas de usuarios del sistema, incluyendo creación, modificación y asignación de roles a aquellas cuentas pertenecientes al tenant.
- Centro de notificaciones: Panel para gestionar y revisar todas las notificaciones del sistema referentes a su rol dentro de la mensajería.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Gestión de perfil: Configuración de información personal.

7.3.3.2 Rol de operador. Los operadores tienen acceso a las funcionalidades operativas esenciales, pero con ciertas restricciones en comparación con los administradores:

- Arqueos de caja: Acceso a la funcionalidad de arqueos, pero sin capacidad de generar reportes detallados ni de cerrar los mismo.
- Gestión de clientes: Capacidades completas para gestionar la información de clientes, hacer su edición, creación y eliminación junto a la asignación de descuento.
- Gestión de pedidos: Acceso total para el manejo de pedidos y seguimiento de entregas.
- Gestión de tarifas: Visualización y aplicación de tarifas existentes.
- Gestión de mensajeros: Coordinación y seguimiento de la actividad de mensajeros.
- Centro de notificaciones: Acceso a notificaciones relevantes para sus funciones.
- Gestión de perfil: Edición de su perfil personal.

7.3.3.3 Rol de mensajero. Los mensajeros tienen acceso limitado a las funcionalidades que son directamente relevantes para su trabajo de campo:

- Pedidos: Visualización de pedidos asignados con información detallada para realizar las entregas, pero sin capacidad de modificación, eliminación o creación.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Centro de notificaciones: Recepción de notificaciones relacionadas con sus asignaciones y actividades.
- Gestión de perfil: Administración de información personal de la cuenta.

7.3.4 Funcionalidades principales

El sistema de mensajería está estructurado en módulos especializados que cubren todas las necesidades operativas y administrativas de una empresa de mensajería. Cada ítem presentado a continuación representa una de estas características principales, diseñadas para optimizar los procesos y garantizar la eficiencia operativa del sistema.

7.3.4.1 Autenticación y gestión de sesiones. El sistema implementa un robusto sistema de autenticación que distingue entre los dos frontends y los diferentes roles de usuario. Para que cualquier usuario pueda ejecutar operaciones en el servidor Web es necesario que en primer lugar inicie sesión a través de una vista especializada que solicita las credenciales apropiadas.

El sistema maneja diferentes tipos de usuarios con permisos específicos, implementando rutas privadas que garantizan que solo los usuarios autenticados y autorizados puedan acceder a las funcionalidades correspondientes a su rol. La gestión de sesiones incluye tokens de seguridad, expiración automática y renovación de sesiones para mantener la seguridad del sistema.

7.3.4.2 Gestión de empresas de mensajería (tenants). En el frontend de gestión de tenants, los superadministradores pueden crear nuevas empresas de mensajería mediante un formulario comprehensivo que incluye información empresarial, configuraciones iniciales y

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

parámetros operativos. El sistema incluye validaciones en tiempo real para asegurar la integridad de los datos y el cumplimiento de reglas de negocio.

Para la modificación de información de tenants existentes, el sistema proporciona formularios pre-poblados con la información actual, permitiendo actualizaciones granulares con notificaciones inmediatas sobre el éxito o fallo de las operaciones.

7.3.4.3 Gestión de clientes. El sistema permite a los usuarios autorizados gestionar la información de los clientes que utilizan los servicios de mensajería. Los usuarios pueden crear nuevos clientes proporcionando información como nombre, dirección, teléfono y correo electrónico. La interfaz incluye validaciones en tiempo real para asegurar que la información ingresada sea correcta y cumpla con los formatos establecidos.

Para la visualización de clientes, el sistema presenta una lista organizada que permite búsqueda y filtrado avanzado, facilitando la localización rápida de clientes específicos, o si lo prefiere una visualización un poco más detallada en modalidad de tarjetas. La funcionalidad de edición permite actualizar la información de los clientes existentes, ya sea tipos de contacto o direcciones relacionadas a este. Se implementó una asignación rápida de direcciones a clientes no existentes basada en pedidos cuyos clientes no están registrados en el sistema al momento de la creación del pedido con la finalidad de facilitar la creación de un perfil de cliente que frecuenta a la empresa.

7.3.4.4 Gestión de pedidos. La funcionalidad de gestión de pedidos constituye el núcleo del sistema de mensajería. Los usuarios autorizados pueden crear nuevos pedidos especificando información detallada como el remitente, destinatario, descripción del paquete, peso, valor declarado y tipo de servicio, entre otros. El formulario de creación de pedidos incluye

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

validaciones comprehensivas que aseguran que toda la información necesaria esté completa antes de procesar el pedido.

El sistema permite el seguimiento en tiempo real del estado de los pedidos, desde su creación hasta su entrega final. Los usuarios pueden visualizar el estado actual de cada pedido, el historial de movimientos y la ruta del envío. La interfaz proporciona filtros avanzados para buscar pedidos por diferentes criterios como fecha, estado, cliente o mensajero junto a validaciones como el correcto flujo de estados del pedido.

Cuenta con la función de precargar las direcciones predeterminadas por clientes, realizar una carga rápida de la información de los pedidos recientes de este cliente con el fin de generar una rápida atención para obtener fidelidad y la generación de un mapa para mostrar la ubicación de recogida y entrega del pedido.

7.3.4.5 Gestión de usuarios. La gestión de usuarios está diseñada específicamente para que los administradores puedan gestionar eficientemente su equipo de trabajo. El sistema permite crear nuevos usuarios del sistema asignándoles roles específicos según sus responsabilidades operativas, editarlos para actualizar información personal, permisos y privilegios, e incluso eliminarlos cuando sea necesario, todo esto respetando los límites establecidos por el plan de suscripción contratado.

La funcionalidad incluye un sistema robusto de restablecimiento de credenciales que permite a los administradores generar nuevas contraseñas temporales para usuarios que han perdido acceso a sus cuentas, garantizando la continuidad operativa del sistema. El módulo implementa validaciones de seguridad que aseguran que solo personal autorizado pueda acceder

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

a estas funcionalidades críticas, manteniendo la integridad del sistema y la información empresarial.

7.3.4.6 Gestión de mensajeros. La gestión de mensajeros es utilizada por administradores y operadores para supervisar y optimizar el rendimiento del equipo de entrega. El sistema permite verificar la disponibilidad de cada mensajero en tiempo real, consultando su estado actual y carga de trabajo asignada. Esta información es crucial para la asignación eficiente de pedidos y el mantenimiento de los tiempos de entrega prometidos.

El módulo incluye un sistema de seguimiento del rendimiento diario que registra el número de entregas completadas. Los usuarios autorizados pueden editar la información del vehículo de trabajo de cada mensajero, asegurando que la información esté actualizada para una correcta planificación logística.

7.3.4.7 Gestión de arqueos de caja. El sistema de arqueos de caja está diseñado para proporcionar un control financiero preciso con diferentes niveles de acceso según el rol del usuario. Los administradores tienen acceso total a todas las funcionalidades, mientras que los operadores cuentan con acceso parcial a funciones específicas de su competencia operativa.

El sistema permite crear un máximo de 3 arqueos por día, correspondientes a los turnos de mañana, tarde y noche, facilitando un control financiero distribuido y detallado. Los ingresos por entrega de pedidos se registran automáticamente en el sistema, pero también se pueden registrar manualmente ingresos de otros tipos o egresos según las necesidades operativas.

Una característica distintiva es que únicamente los administradores pueden cerrar arqueos después de verificarlos exhaustivamente y generar reportes detallados de estos, garantizando la

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

supervisión adecuada y el control financiero del negocio. Este flujo de trabajo asegura la transparencia y trazabilidad de todas las transacciones financieras.

7.3.4.8 Dashboard y estadísticas. El dashboard está diseñado específicamente para que los administradores puedan observar las métricas del día actual de manera integral y en tiempo real, métricas semanales, mensuales o en un período personalizado. La interfaz presenta información clave como la cantidad de pedidos activos en general, cuántos se entregaron en el día, cuántos faltan por completar, proporcionando una visión clara del estado operativo.

Las métricas financieras incluyen el ingreso total del día y el ingreso promedio por pedido, permitiendo a los administradores evaluar el rendimiento económico y tomar decisiones informadas sobre precios y estrategias comerciales. El sistema también presenta el desempeño de los mensajeros en tiempo real, incluyendo entregas completadas, tiempo de respuesta y eficiencia.

Esta centralización de información permite a los administradores identificar tendencias, detectar problemas operativos y optimizar recursos de manera proactiva, mejorando la eficiencia general del servicio de mensajería.

7.3.4.9 Gestión de tarifas. La gestión de tarifas es accesible tanto para administradores como para operadores, quienes tienen control sobre los precios a aplicar según las características específicas de cada pedido. El sistema permite establecer tarifas base según la necesidad de cada empresa en diferentes variables como peso, dimensiones, distancia de entrega, tipo de servicio (normal, express, urgente) y zonas geográficas.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Los usuarios autorizados pueden crear, editar y activar diferentes estructuras tarifarias, permitiendo flexibilidad en la estrategia de precios según las necesidades del mercado.

7.3.4.10 Gestión de notificaciones. El sistema de notificaciones opera de manera diferenciada según el tipo de usuario, adaptándose a las necesidades específicas de cada rol. Principalmente, las notificaciones se utilizan para informar sobre pedidos y cambios en su estado, manteniendo a todos los usuarios informados sobre el progreso de las entregas.

Para los administradores, el sistema genera notificaciones adicionales cuando se detectan arqueos con diferencias financieras, permitiendo una investigación a fondo de posibles discrepancias y manteniendo la integridad del control financiero.

7.3.4.11 Interfaz responsiva y experiencia de usuario. Ambas aplicaciones están diseñadas con principios de diseño responsivo utilizando Bootstrap, garantizando que las interfaces se adapten correctamente a diferentes tamaños de pantalla y dispositivos. Esto permite que los usuarios puedan acceder al sistema desde computadores de escritorio, o en un futuro a tabletas o dispositivos móviles manteniendo una experiencia de usuario consistente y optimizada.

7.3.5 Estructura de componentes

La arquitectura de ambos frontends está organizada en una estructura de componentes modulares que facilita el mantenimiento y la escalabilidad de las aplicaciones. Cada componente se conforma de elementos gráficos que conforman la vista de la interfaz de usuario y de las operaciones que se ejecutan con cada interacción del usuario con los elementos de la interfaz.

7.3.5.1 Componentes del frontend de tenants.

- Componentes de autenticación para superadministradores.
- Componentes de gestión de tenants.
- Componentes de administración de administradores de mensajería.
- Componentes de navegación y rutas protegidas.

7.3.5.5 Componentes del frontend de mensajería.

- Componentes de autenticación y gestión de sesiones multi-rol.
- Componentes de gestión de empresas.
- Componentes de gestión de clientes.
- Componentes de gestión de pedidos.
- Componentes de gestión de mensajeros.
- Componentes de arqueos de caja.
- Componentes de dashboard y estadísticas.
- Componentes de gestión de tarifas.
- Componentes de gestión de usuarios.
- Componentes de notificaciones.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Componentes de navegación y rutas protegidas por rol.
- Componentes de visualización de datos y reportes especializados.

7.3.6 Integración con el backend

La integración entre los frontends y el backend del sistema de mensajería express implementa una arquitectura distribuida que garantiza la seguridad, escalabilidad y eficiencia del sistema multi-tenant. Esta integración se basa en la comunicación a través de servicios RESTful y un sistema de proxy que actúa como mediador entre la aplicación frontend de los inquilinos y el microservicio backend de los inquilinos.

7.3.6.1 Arquitectura de comunicación.

7.3.6.1.1 Frontend de mensajería proxy-service. El frontend de mensajería no se comunica directamente con el microservicio de mensajería, sino que utiliza un proxy service como intermediario. Esta arquitectura proporciona una capa adicional de seguridad y control de acceso que es fundamental para el funcionamiento del sistema multi-tenant.

Flujo de comunicación:

1. Solicitud del cliente: El frontend de mensajería envía peticiones HTTP a través de Axios hacia el proxy service.
2. Validación de tenant: El proxy service intercepta la petición y valida el estado del tenant consultando el microservicio de tenants.
3. Verificación de estado: Se verifica que el tenant esté activo y tenga permisos para realizar la operación solicitada.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

4. Enrutamiento condicional:

- a. Si el tenant está activo: La petición se reenvía al microservicio de mensajería.
- b. Si el tenant está inactivo: Se retorna un error de acceso denegado.

5. Respuesta del sistema: La respuesta del microservicio se envía de vuelta al frontend a través del proxy.

7.3.6.1.2 Frontend de tenants comunicación directa. El frontend de gestión de tenants tiene un patrón de comunicación más directo, pero con múltiples endpoints según la funcionalidad requerida.

Comunicación con el microservicio de tenants:

- Gestión de empresas de mensajería (CRUD de tenants).
- Monitoreo del estado de los tenants.
- Gestión de superadministradores.

Comunicación con el microservicio de mensajería:

- Consulta de usuarios administradores para autenticación.
- Validación de credenciales de administradores.
- Sincronización de información de usuarios.

7.3.6.2 Sistema de validación y seguridad.

7.3.6.2.1 Validación de tenants. El proxy service implementa un sistema robusto de validación que incluye:

Verificación de estado activo:

- Consulta en tiempo real del estado del tenant en la base de datos.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Validación de fechas de suscripción y límites de uso.
- Verificación de permisos específicos para cada operación.

Caché de validaciones:

- Implementación de cache temporal para reducir latencia.
- Actualización automática del cache cuando cambia el estado del tenant.
- Invalidación inmediata en caso de desactivación del tenant.

7.3.6.2.2 Control de accesos. El sistema implementa múltiples niveles de control de acceso:

Autenticación de usuarios:

- Validación de tokens JWT en cada petición.
- Verificación de roles y permisos específicos.

Autorización por recursos:

- Validación de permisos específicos para cada endpoint.
- Verificación de ownership de recursos (datos pertenecientes al tenant).
- Implementación de políticas de acceso granulares.

7.3.6.3 Manejo de peticiones HTTP.

7.3.6.3.1 Configuración de axios. Ambos frontends utilizan Axios con configuraciones específicas para el manejo de peticiones:

Interceptores de peticiones:

- Adición automática de headers de autenticación.
- Configuración de timeouts apropiados.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Manejo de headers específicos para identificación de tenant.

Interceptores de respuestas:

- Manejo centralizado de errores HTTP.

7.3.6.3.2 Manejo de errores. El sistema implementa un manejo comprehensivo de errores:

Errores de autenticación:

- Redirección automática a pantallas de login.
- Limpieza de tokens inválidos.
- Notificaciones claras sobre problemas de acceso.

Errores de tenant inactivo:

- Mensajes específicos sobre estado del tenant.
- Bloqueo temporal de funcionalidades.

7.3.6.3.3 Consistencia de datos. El sistema garantiza la consistencia de datos a través de:

Validación de integridad:

- Verificación de datos antes de envío al backend.
- Validación de formatos y tipos de datos.
- Verificación de reglas de negocio en el frontend.

7.4 Uso de datos

Para permitir a las empresas de mensajería gestionar eficientemente sus operaciones de entrega, la aplicación consume datos de múltiples fuentes relacionadas con clientes, pedidos, mensajeros y operaciones logísticas. Estos datos son proporcionados tanto por las propias empresas de mensajería como por sistemas externos de integración, requiriendo un procesamiento previo para garantizar la calidad y consistencia de la información en un entorno multitenant.

7.4.1 Arquitectura multitenant

El sistema implementa un modelo de aislamiento híbrido con dos esquemas principales encontrados en dos bases de datos diferentes.

7.4.1.1 Esquema tenant. Centraliza la información de configuración y administración de tenants.

- Gestión de planes: Control de límites por empresa (usuarios, pedidos mensuales, pedidos simultáneos).
- Estados de tenant: Seguimiento del estado operativo de cada empresa.
- Administración centralizada: Información de contacto y configuración global.

7.4.1.2 Esquema mensajería. Contiene todos los datos operacionales con aislamiento por tenant_id.

- Aislamiento de datos: Cada tenant accede únicamente a su información mediante filtros por tenant_id.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Escalabilidad: Capacidad de incorporar nuevas empresas sin afectar las existentes.
- Seguridad: Prevención de acceso cruzado entre diferentes empresas de mensajería.
- Auditoría: Trazabilidad completa de cambios mediante historial de pedidos.

7.4.2 Procesamiento de datos

7.4.2.1 Fase de perfilamiento general. Previo a la carga de información, es necesario realizar un análisis exhaustivo de los datos recibidos considerando la estructura multitenant:

Estructura de datos de empresas de mensajería:

- Información corporativa (nombre, dirección, contactos).
- Configuración de planes y límites operacionales.
- Tarifas y tipos de servicio específicos por empresa.

Estructura de datos de usuarios:

- Roles específicos (admin_mensajería, operador, mensajero).
- Asociación con empresa de mensajería.
- Credenciales y permisos por tenant.

Información de mensajeros:

- Disponibilidad y capacidad de carga.
- Tipo de vehículo y límites de pedidos simultáneos.
- Métricas de rendimiento.

Datos de pedidos:

- Direcciones de recogida y entrega (temporales y permanentes).
- Estados del flujo de entrega y tiempos de procesamiento.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Información de tarifas y costos por empresa.

Información de clientes:

- Datos de contacto y direcciones frecuentes.
- Historial de pedidos y descuentos aplicables.
- Frecuencia de uso por empresa de mensajería.

Datos de arqueo de caja:

- Ingresos por turno y fecha.
- Diferencias de caja y seguimiento financiero.
- Tipos de ingreso automáticos y manuales.

7.4.2.2 Fase de limpieza de datos. Durante el proceso de limpieza se eliminan de los conjuntos de datos todas las columnas que no sean de interés para las operaciones de mensajería:

Criterios de limpieza aplicados:

- Eliminación de duplicados:
 - Usuarios con combinaciones repetidas de email y tenant_id.
 - Direcciones duplicadas dentro del mismo tenant.
 - Pedidos con información redundante.
- Normalización de direcciones:
 - Estandarización de ciudades y barrios.
 - Validación de direcciones completas.
 - Clasificación entre direcciones de recogida y entrega.
- Validación de contactos:
 - Verificación de formatos de números telefónicos (10-15 dígitos).

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Validación de direcciones de correo electrónico.
- Limpieza de caracteres especiales.
- Filtrado de datos obsoletos:
 - Eliminación de pedidos con estados inconsistentes.
 - Validación de fechas de creación y entrega.
 - Limpieza de registros con tenant_id inválidos.

7.4.2.3 Fase de validación regla de negocios. Posterior a la limpieza, los datos continúan a la fase de validación de reglas de negocio específicas del dominio de mensajería:

7.4.2.3.1 Reglas de tenants y empresas.

- El nombre de la empresa debe ser único por tenant.
- Cada tenant debe tener al menos un plan asignado.
- Los límites del plan deben ser coherentes con la operación.
- El admin_mensajería debe existir y tener el rol correspondiente.

7.4.2.3.2 Reglas para usuarios.

- La combinación de email y tenant_id debe ser única.
- La combinación de nombre_usuario y tenant_id debe ser única.
- Los usuarios con rol 'mensajero' deben tener un registro correspondiente en la tabla mensajero.
- Las contraseñas deben cumplir con políticas de seguridad establecidas.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Los roles deben corresponder a valores válidos en rol_usuario.
- Todos los usuarios deben estar asociados a un tenant válido.

7.4.2.3.3 Reglas para clientes.

- Los números telefónicos deben seguir formatos válidos (10-15 dígitos).
- Debe existir al menos un método de contacto válido.
- La frecuencia de pedidos debe ser coherente con el historial.
- Los descuentos deben estar en el rango 0-100%.
- Cada cliente debe estar asociado a una empresa de mensajería válida.

7.4.2.3.4 Reglas para pedidos.

- Los pedidos deben tener direcciones de recogida y entrega válidas.
- El tipo de servicio debe corresponder a categorías predefinidas.
- Los estados deben seguir el flujo: 'PENDIENTE' → 'ASIGNADO' → 'EN_TRANSITO' → 'ENTREGADO'.
- Un mensajero solo puede tener pedidos activos según su límite máximo.
- Las fechas de creación no pueden ser futuras.
- Los costos y totales deben ser coherentes con las tarifas aplicadas.
- Los teléfonos de recogida y entrega son obligatorios.

7.4.2.3.5 Reglas para mensajeros.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Solo usuarios con rol 'mensajero' pueden ser registrados como mensajeros.
- Un mensajero no disponible no puede recibir nuevas asignaciones.
- Los pedidos activos no pueden exceder el límite máximo configurado.
- El tipo de vehículo debe ser válido.
- La disponibilidad debe actualizarse automáticamente según asignaciones.

7.4.2.3.6 Reglas para direcciones.

- Las direcciones deben tener ciudad y dirección completa.
- La clasificación de recogida/entrega debe ser coherente.
- No pueden existir direcciones duplicadas por tenant.
- Las direcciones asociadas a clientes deben mantener consistencia.

7.4.2.3.7 Reglas para tarifas.

- Cada empresa debe tener al menos una tarifa activa.
- Los valores fijos deben ser positivos.
- No pueden existir tarifas con nombres duplicados por empresa.

7.4.2.3.8 Reglas para arqueos de caja.

- Solo puede existir un arqueo por fecha, turno y empresa.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Los montos deben ser coherentes con los ingresos registrados.
- Las diferencias calculadas deben ser automáticas.
- Los ingresos automáticos por pedidos entregados deben registrarse.

7.4.2.3.9 Reglas para historial de pedido.

- Todos los cambios en pedidos deben quedar registrados.
- Los tipos de cambio deben corresponder a campos modificables.
- Los valores anteriores y nuevos deben ser coherentes.
- El usuario que realiza el cambio debe ser válido.

7.4.3 Validación y persistencia.

Estas verificaciones de integridad de los datos ocurren en múltiples capas:

7.4.3.1 Validación a nivel de base de datos.

- Triggers automáticos: Mantienen la consistencia de datos relacionados.
- Restricciones de integridad: Foreign keys y constraints únicos.
- Validación de tipos: Tipos de datos y rangos válidos.

7.4.3.2 Validación a nivel de aplicación.

- Reglas de negocio: Implementadas en la lógica de la aplicación.
- Validación de límites: Control de planes y restricciones por tenant.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Sanitización de datos: Limpieza previa a la persistencia.

7.4.3.3 Tipos de datos soportados.

- Datos de carga masiva: Información de sistemas externos o procesos batch.
- Datos de interfaz web: Capturados directamente por usuarios del sistema.
- Datos de integración: APIs y webhooks de terceros.

7.4.4 Triggers y automatización.

El sistema implementa triggers automáticos para mantener la consistencia:

7.4.4.1 Gestión de mensajeros.

- Creación automática: Registro de mensajero al asignar rol correspondiente.
- Sincronización de estados: Mantenimiento de consistencia entre usuario y mensajero.
- Actualización de disponibilidad: Basada en pedidos activos y límites.

7.4.4.2 Gestión de pedidos.

- Cálculo automático de totales: Aplicación de tarifas y descuentos.
- Actualización de contadores: Pedidos activos por mensajero.
- Registro de historial: Trazabilidad de todos los cambios.

7.4.4.3 Gestión de arqueos de caja.

- Ingresos automáticos: Registro por pedidos entregados.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Actualización de totales: Recálculo al modificar ingresos.
- Validación de diferencias: Cálculo automático de discrepancias.

7.4.4.4 Gestión de clientes.

- Actualización de frecuencia: Contador de pedidos por cliente.
- Fecha de último pedido: Seguimiento de actividad.

7.4.5 Índices y optimización.

Para garantizar el rendimiento en un entorno multitenant, se implementan índices estratégicos:

7.4.5.1 Índices por tenant.

- Todas las tablas principales incluyen índices en tenant_id.
- Índices compuestos para consultas frecuentes por tenant.

7.4.5.2 Índices operacionales.

- Estados y fechas de pedidos para dashboards.
- Disponibilidad de mensajeros para asignaciones.
- Clientes y direcciones para búsquedas rápidas.

7.4.5.3 Índices de rendimiento.

- Métricas de mensajeros y rankings.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Arqueos por fecha y turno.
- Historial de pedidos por fecha.

7.4.6 Vistas y reportes.

El sistema incluye vistas optimizadas para análisis:

7.4.6.1 Vistas de estadísticas.

- `v_estadisticas_mensajeros`: Métricas de rendimiento por mensajero.
- `v_ranking_mensajeros`: Clasificación y categorización de desempeño.
- `v_dashboard_general`: Indicadores operacionales por fecha.

7.4.6.2 Vistas de gestión.

- `v_tenant_info`: Información completa de tenants y planes.
- `v_resumen_arqueos`: Consolidado financiero por período.

Al concluir con la validación de las reglas de integridad y la aplicación de las transformaciones necesarias, los datos resultantes son cargados al servidor de base de datos correspondiente a cada tenant, garantizando el aislamiento, la integridad y la optimización de la información por empresa de mensajería.

8. Implementación del prototipo

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

La implementación del prototipo del sistema de mensajería express requirió seleccionar y configurar diversas herramientas tecnológicas que permitieran desarrollar una solución robusta, escalable y mantenible, acorde con la arquitectura diseñada. A continuación, se detallan las principales herramientas utilizadas en este proceso.

8.1 Backend Spring Boot

Para el desarrollo del backend se utilizó Spring Boot 3.5.3 con Java 17, implementando dos servicios independientes que forman parte de la arquitectura de microservicios. La elección de Spring Boot se fundamenta en las siguientes características:

8.1.1 Dependencias principales

- Spring Boot Starter Web: Proporciona la funcionalidad necesaria para desarrollar aplicaciones web RESTful, incluyendo el servidor embebido Apache Tomcat y las librerías necesarias para el manejo de peticiones HTTP.
- Spring Boot Starter Data JPA: Facilita la implementación de la capa de persistencia mediante el patrón Repository, proporcionando abstracción sobre las operaciones de base de datos y soporte para consultas personalizadas.
- Spring Boot Starter Security: Implementa la seguridad de la aplicación mediante autenticación y autorización, integrándose con el sistema de tokens JWT para el manejo de sesiones stateless.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

8.1.2 Autenticación y autorización

La seguridad del sistema se implementó utilizando JSON Web Tokens (JWT) mediante las librerías:

- JJWT API 0.11.5: Proporciona las interfaces necesarias para la creación y validación de tokens JWT.
- JJWT Implementation 0.11.5: Contiene la implementación concreta de las funcionalidades JWT.
- JJWT Jackson 0.11.5: Facilita la serialización y deserialización de los tokens en formato JSON.

8.1.3 Documentación de apis

- SpringDoc OpenAPI 2.3.0: Genera automáticamente la documentación de las APIs REST utilizando el estándar OpenAPI 3.0, proporcionando una interfaz interactiva para pruebas y documentación técnica.

8.1.2 Generación de reportes

- iText PDF 5.5.13.3: Librería para la generación de documentos PDF, utilizada para crear reportes y documentos oficiales en el sistema de mensajería.
- Apache POI 5.4.1: Conjunto de librerías para el manejo de documentos Microsoft Office, específicamente para la generación de reportes en formato Excel y CSV.

8.2 Frontend React-Vite

El frontend se desarrolló utilizando React 19.1.0 con Vite como herramienta de construcción, implementando dos aplicaciones cliente independientes que se conectan a sus respectivos servicios backend.

8.2.1 Herramientas de desarrollo

- Vite 6.3.5: Herramienta de construcción moderna que proporciona un servidor de desarrollo rápido y un sistema de construcción optimizado para aplicaciones web modernas.
- React 19.1.0: Biblioteca de JavaScript para la construcción de interfaces de usuario, proporcionando un enfoque declarativo y basado en componentes.

8.2.2 Comunicación HTTP

- Axios 1.9.0: Cliente HTTP para realizar peticiones a las APIs REST del backend, proporcionando interceptores para el manejo de tokens JWT y manejo de errores centralizado.

8.2.3 Interfaz de usuario

- Bootstrap 5.3.6: Framework CSS que proporciona componentes predefinidos y un sistema grid responsivo para crear interfaces de usuario consistentes y profesionales.
- Lucide React 0.525.0: Librería de iconos vectoriales optimizada para React, proporcionando iconografía moderna y escalable.

8.2.4 Navegación y enrutamiento

- React Router DOM 7.6.0: Librería de enrutamiento para aplicaciones React de una sola página (SPA), permitiendo la navegación entre diferentes vistas sin recargar la página.
- React Router Hash Link 2.4.3: Extensión de React Router que permite navegación directa a secciones específicas de una página mediante anchors.

8.2.5 Visualización de datos

- Recharts 3.0.2: Librería de gráficos construida específicamente para React, utilizada para crear visualizaciones de datos interactivas y responsivas.

Figura 31

Distribución de pedidos gráfica torta



8.2.6 Mapas interactivos

- Leaflet 1.9.4: Librería de mapas interactivos de código abierto, utilizada para mostrar ubicaciones y rutas en el sistema de mensajería.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

8.3.1 Características principales

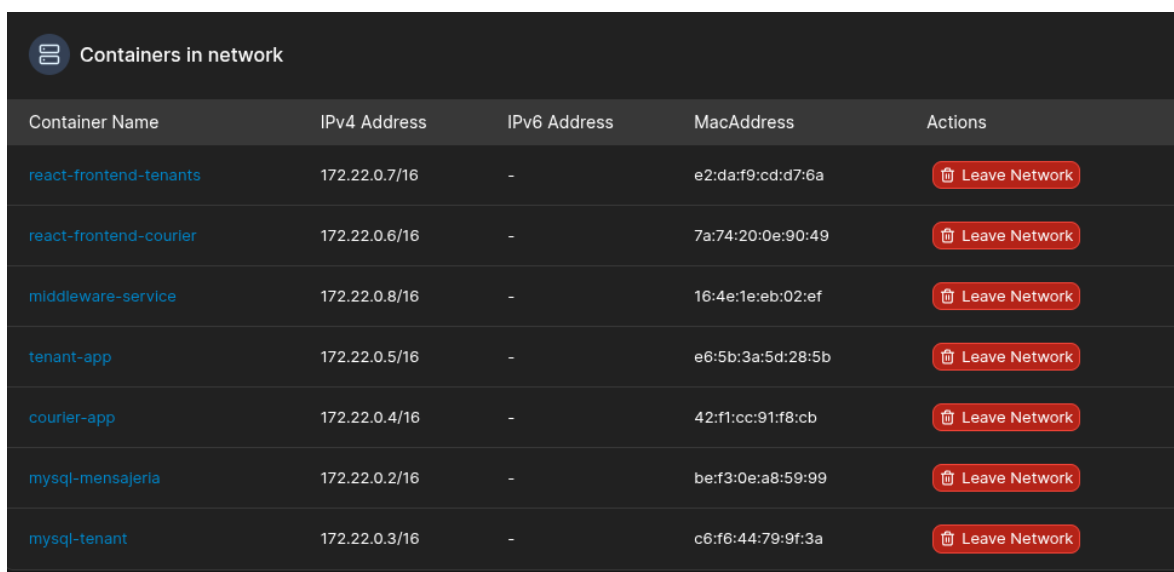
- Compatibilidad ACID: Garantiza la consistencia e integridad de los datos.
- Escalabilidad: Soporte para grandes volúmenes de datos y múltiples conexiones concurrentes.
- Replicación: Capacidad de configurar réplicas para alta disponibilidad.
- Índices optimizados: Mejora del rendimiento en consultas multitenant.

8.4 Contenedor Docker

El sistema completo se implementó en contenedores Docker, garantizando así la consistencia del entorno de desarrollo y facilitando su despliegue en diferentes ambientes, tanto locales como en la nube.

Figura 33

Contenedor docker



The screenshot shows the 'Containers in network' view in Docker Desktop. It displays a table with the following columns: Container Name, IPv4 Address, IPv6 Address, MacAddress, and Actions. There are seven containers listed, each with a 'Leave Network' button in the Actions column.

Container Name	IPv4 Address	IPv6 Address	MacAddress	Actions
react-frontend-tenants	172.22.0.7/16	-	e2:da:f9:cd:d7:6a	Leave Network
react-frontend-courier	172.22.0.6/16	-	7a:74:20:0e:90:49	Leave Network
middleware-service	172.22.0.8/16	-	16:4e:1e:eb:02:ef	Leave Network
tenant-app	172.22.0.5/16	-	e6:5b:3a:5d:28:5b	Leave Network
courier-app	172.22.0.4/16	-	42:f1:cc:91:f8:cb	Leave Network
mysql-mensajeria	172.22.0.2/16	-	be:f3:0e:a8:59:99	Leave Network
mysql-tenant	172.22.0.3/16	-	c6:f6:44:79:9f:3a	Leave Network

8.5 Proxy Node.js

El proxy implementado consiste en un servidor Node.js que utiliza la librería Express para gestionar el enrutamiento de peticiones entre los distintos servicios de la arquitectura multitenant, proporcionando un único punto de entrada para las aplicaciones cliente (Inquilinos).

Figura 34

Función del proxy que valida datos del tenant

```
proxy-middleware-service

1 async function validateWithTenant(additionalData, authToken) {
2   const {tenantId} = additionalData;
3
4   try {
5     const response = await axios.get(`${TENANT_SERVICE_URL}/api/tenants/${tenantId}/info`, {
6       headers: {
7         'Authorization': `${authToken}`,
8         'Content-Type': 'application/json'
9       },
10      timeout: 10000 // 10 segundos timeout
11    });
12
13    // condicional que valida si la informacion del tenant que se devuelve esta activo
14
15    console.log('Respuesta validacion en server.js -> ', response.data.estado);
16    if(response.data.estado == 'ACTIVO') {
17      console.warn('SI ESTA ACTIVO EL TENANT')
18      return true;
19    } else {
20      return false;
21    }
22  } catch (error) {
23    if (error.response) {
24      // La petición fue hecha y el servidor respondió con un código de error
25      console.error('Error desde el servicio tenant:', error.response.status);
26    } else if (error.request) {
27      // La petición fue hecha pero no se recibió respuesta
28      console.error('No se recibió respuesta del servicio tenant');
29    } else {
30      // Error al configurar la petición
31      console.error('Error al configurar la petición:', error.message);
32    }
33    return false;
34  }
35 }
36 }
```

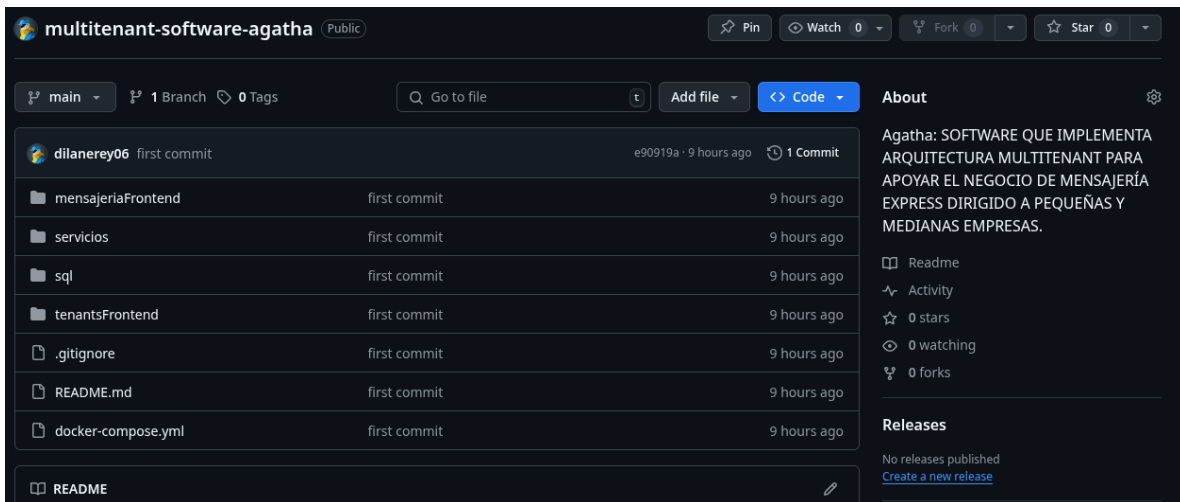
SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

8.6 GitHub

Como sistema de control de versiones y repositorio central de código se utilizó GitHub (en su versión web más reciente). El repositorio sigue la estructura de ramas Git estándar, con la rama 'main' como línea base de desarrollo.

Figura 35

Repositorio github



8.7 Conclusiones del stack tecnológico

La combinación de tecnologías seleccionadas proporciona una base sólida para la implementación de la arquitectura multitenant, ofreciendo:

1. Escalabilidad horizontal: Capacidad de agregar nuevos inquilinos sin impacto en el rendimiento.
2. Mantenibilidad: Separación clara de responsabilidades entre servicios.
3. Seguridad: Implementación robusta de autenticación y autorización.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

4. Portabilidad: Facilidad de despliegue en diferentes entornos mediante el uso de contenedores.
5. Monitoreo: Capacidad de seguimiento y análisis del comportamiento del sistema.

Esta arquitectura tecnológica permite satisfacer los requisitos funcionales y no funcionales del sistema de mensajería express, proporcionando una base escalable para el crecimiento futuro del proyecto.

9. Validación y pruebas funcionales

Para garantizar el correcto funcionamiento del sistema de gestión de mensajería, se desarrolló un plan integral de pruebas funcionales compuesto por 21 casos de prueba que cubren todas las funcionalidades principales del sistema. Estos casos de prueba fueron diseñados para validar tanto los flujos principales de trabajo como los casos excepcionales y de error, asegurando la robustez y confiabilidad de la aplicación.

9.1 Plan de pruebas

9.1.1 CP-001: Creación de pedido

Prerrequisitos: Ser usuario con rol administrador u operador, tener sesión activa, tener clientes y mensajeros registrados.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Descripción: Ingresar a la sección de pedidos y crear un nuevo pedido ingresando todos los datos requeridos (cliente, dirección de recogida, dirección de entrega, descripción del paquete, peso, valor declarado, tipo de servicio, etc.).

Ejecución:

1. Iniciar sesión como administrador u operador.
2. Navegar a la sección "Pedidos".
3. Hacer clic en "Nuevo Pedido".
4. Completar el formulario con información válida.
5. Confirmar creación del pedido.

Resultado: El pedido se crea correctamente con estado "PENDIENTE", se genera un ID único, se registra en el historial y se muestra en la lista de pedidos.

9.1.2 CP-002: Asignación de mensajero a pedido

Prerrequisitos: Tener pedido en estado "PENDIENTE", tener mensajeros disponibles, ser administrador u operador.

Descripción: Asignar un mensajero disponible a un pedido existente para proceder con la entrega.

Ejecución:

1. Acceder a la sección de pedidos.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

2. Seleccionar un pedido en estado "PENDIENTE" con el botón de edición.
3. Ingresar el nombre del mensajero en el campo "Mensajero".
4. Seleccionar un mensajero disponible de la lista.
5. Confirmar la asignación.

Resultado: El pedido cambia a estado "ASIGNADO", el mensajero queda vinculado al pedido, se actualiza su disponibilidad y se genera notificación automática.

9.1.3 CP-003: Registro de nuevo cliente

Prerrequisitos: Ser administrador u operador, tener sesión activa.

Descripción: Crear un nuevo cliente en la base de datos con información completa de contacto.

Ejecución:

1. Navegar a "Clientes".
2. Hacer clic en "Nuevo Cliente".
3. Ingresar datos correspondientes.
4. Guardar información.

Resultado: El cliente se registra correctamente, aparece en la lista de clientes y queda disponible para asignación en pedidos.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

9.1.4 CP-004: Creación de arqueo de caja

Prerrequisitos: Ser administrador u operador, no tener arqueo activo en el turno actual.

Descripción: Crear un nuevo arqueo de caja para controlar ingresos y egresos del turno.

Ejecución:

1. Acceder a "Arqueos de caja".
2. Seleccionar "Nuevo Arqueo".
3. Especificar turno (mañana, tarde o noche).
4. Ingresar monto inicial de caja.
5. Confirmar creación.

Resultado: Se crea el arqueo correctamente, queda en estado activo y listo para registrar movimientos financieros.

9.1.5 CP-005: Edición de información mensajero

Prerrequisitos: Tener mensajeros registrados, ser administrador u operador.

Descripción: Modificar información del vehículo y disponibilidad de un mensajero existente.

Ejecución:

1. Ir a "Mensajeros".

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

2. Seleccionar un mensajero de la lista.
3. Hacer clic en "Editar".
4. Modificar tipo de vehículo.
5. Cambiar estado de disponibilidad.
6. Guardar cambios.

Resultado: La información del mensajero se actualiza correctamente y los cambios se reflejan en el sistema de asignación.

9.1.6 CP-006: Búsqueda y filtrado de pedidos

Prerrequisitos: Tener pedidos registrados en el sistema, ser usuario autorizado.

Descripción: Utilizar los filtros para buscar pedidos específicos por diferentes criterios.

Ejecución:

1. Acceder a la lista de pedidos.
2. Usar filtro por fecha de creación.
3. Filtrar por estado del pedido.
4. Buscar por nombre de cliente.
5. Aplicar filtro por mensajero asignado.
6. Revisar resultados.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Resultado: El sistema muestra únicamente los pedidos que cumplen con los criterios especificados.

9.1.7 CP-007: Generación de reporte de ingresos

Prerrequisitos: Ser administrador, tener ingresos registrados en arqueos de caja.

Descripción: Generar un reporte de ingresos para un período específico.

Ejecución:

1. Acceder al "Arqueo de caja".
2. Seleccionar "Generar reporte".
3. Configurar rango de fechas.
4. Elegir tipo de formato.
5. Generar reporte.
6. Exportar en formato deseado.

Resultado: Se genera correctamente el reporte con ingresos y se exporta el archivo PDF o XLS.

9.1.8 CP-008: Actualización de estado pedido

Prerrequisitos: Tener pedido asignado, ser administrador u operador.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Descripción: Cambiar el estado de un pedido siguiendo el flujo: PENDIENTE → ASIGNADO → EN_TRANSITO → ENTREGADO.

Ejecución:

1. Seleccionar pedido en estado "ASIGNADO".
2. Editar campo "Estado".
3. Cambiar a "EN_TRANSITO".
4. Confirmar cambio de estado.
5. Verificar registro en historial.

Resultado: El estado se actualiza correctamente, se registra en el historial con fecha y usuario, y se generan notificaciones correspondientes.

9.1.9 CP-009: Configuración de tarifas

Prerrequisitos: Ser administrador u operador, tener acceso a gestión de tarifas.

Descripción: Crear una nueva tarifa para servicios de mensajería con diferentes parámetros.

Ejecución:

1. Navegar a "Tarifas".
2. Hacer clic en "Nueva tarifa".

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

3. Ingresar nombre descriptivo.
4. Configurar valor base.
5. Activar tarifa.
6. Guardar configuración.

Resultado: La tarifa se crea y activa correctamente, quedando disponible para aplicar en nuevos pedidos.

9.1.10 CP-010: Cierre de arqueo de caja

Prerrequisitos: Ser administrador, tener arqueo activo con movimientos registrados.

Descripción: Cerrar un arqueo de caja después de verificar ingresos y egresos del turno.

Ejecución:

1. Acceder al arqueo activo.
2. Verificar todos los movimientos registrados.
3. Revisar ingresos automáticos por pedidos entregados.
4. Validar diferencias de caja.
5. Hacer clic en "Cerrar arqueo".
6. Generar reporte final.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Resultado: El arqueo se cierra correctamente, se genera el reporte detallado y no permite más modificaciones.

9.1.11 CP-011: Visualización de dashboard con métricas

Prerrequisitos: Ser administrador, tener datos de pedidos y operaciones del día.

Descripción: Consultar las métricas del día actual en el dashboard principal.

Ejecución:

1. Acceder a "Estadísticas".
2. Revisar pedidos activos del día.
3. Consultar pedidos entregados.
4. Verificar ingresos totales.
5. Analizar rendimiento de mensajeros.

Resultado: El dashboard muestra correctamente todas las métricas actualizadas en tiempo real.

9.1.12 CP-012: Gestión de notificaciones

Prerrequisitos: Tener cambios en pedidos o arqueos con diferencias, ser usuario autorizado.

Descripción: Revisar y gestionar las notificaciones generadas por el sistema.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Ejecución:

1. Acceder a "Notificaciones".
2. Revisar notificaciones pendientes.
3. Marcar notificaciones como leídas.
4. Filtrar por tipo de notificación.
5. Verificar notificaciones de arqueos con diferencias.

Resultado: Las notificaciones se muestran correctamente organizadas y se pueden gestionar adecuadamente.

9.1.13 CP-013: Creación de pedido con datos inválidos

Prerrequisitos: Ser administrador u operador, tener sesión activa.

Descripción: Intentar crear un pedido con información incompleta o inválida para verificar las validaciones del sistema.

Ejecución:

1. Navegar a "Nuevo pedido".
2. Dejar campos obligatorios vacíos (dirección de recogida).
3. Ingresar teléfono con formato inválido (menos de 10 dígitos).
4. Intentar guardar el pedido.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

5. Verificar mensajes de error.

Resultado: El sistema debe rechazar la creación del pedido y mostrar mensajes de error específicos indicando los campos inválidos o faltantes.

9.1.14 CP-014: Creación de pedido anterior de un cliente registrado

Prerrequisitos: Ser administrador u operador, tener un cliente registrado con pedidos entregados.

Descripción: Crear un nuevo pedido a base de un pedido anterior entregado para un cliente específico.

Ejecución:

1. Iniciar sesión como administrador u operador.
2. Verificar que haya un cliente con un pedido entregado.
3. Ir a la sección "Pedidos".
4. Escoger al cliente con pedido anterior.
5. Seleccionar la opción de utilizar información de un pedido antiguo.
6. Guardar pedido.
7. Verificar que efectivamente la información relevante es correcta.

Resultado: El sistema debe completar la solicitud y mostrar el nuevo pedido correctamente a base del pedido entregado.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

9.1.15 CP-015: Creación cuarto arqueo de caja en el mismo día

Prerrequisitos: Tener 3 arqueos ya creados en el día actual (mañana, tarde, noche), ser administrador.

Descripción: Intentar crear un cuarto arqueo en el mismo día para verificar la restricción de máximo 3 arqueos diarios.

Ejecución:

1. Verificar que existen 3 arqueos en el día actual.
2. Acceder a "Arqueos de caja".
3. Intentar crear "Nuevo arqueo".
4. Seleccionar cualquier turno disponible.
5. Intentar confirmar la creación.

Resultado: El sistema debe rechazar la creación del arqueo y mostrar mensaje de error indicando que se ha alcanzado el límite máximo de 3 arqueos por día.

9.1.16 CP-016: Asignación de descuento a cliente frecuente

Prerrequisitos: Tener cliente registrado con historial de pedidos, ser administrador u operador.

Descripción: Asignar un descuento especial a un cliente que cumple con criterios de frecuencia de pedidos.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Ejecución:

1. Acceder a "Clientes".
2. Buscar cliente con múltiples pedidos.
3. Seleccionar "Editar cliente".
4. Ir a sección "Descuento".
5. Ingresar porcentaje de descuento (10%).
6. Guardar cambios.

Resultado: El descuento se asigna correctamente al cliente y se aplicará automáticamente en futuros pedidos.

9.1.17 CP-017: Registro de ingreso manual en arqueo de caja

Prerrequisitos: Tener arqueo activo, ser administrador u operador.

Descripción: Registrar un ingreso manual en el arqueo de caja por concepto diferente a entrega de pedidos.

Ejecución:

1. Acceder al arqueo activo.
2. Hacer clic en "Añadir ingreso Manual".
3. Ingresar concepto (ej: "Venta de material de empaque").

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

4. Especificar monto del ingreso.
5. Agregar descripción detallada.
6. Confirmar registro.

Resultado: El ingreso manual se registra correctamente, se actualiza el total del arqueo y se clasifica como ingreso manual.

9.1.18 CP-018: Creación de tenant y verificación de aislamiento de datos

Prerrequisitos: Ser superadministrador o administrador del sistema.

Descripción: Crear un nuevo tenant con su administrador de mensajería asociado y verificar que el aislamiento de datos funciona correctamente, confirmando que un tenant no tiene acceso a información de otros tenants.

Ejecución:

1. Iniciar sesión en el frontend de manejo de tenants como superadministrador.
2. Navegar a la sección "Tenants".
3. Hacer clic en "Crear nuevo tenant".
4. Completar el formulario con datos válidos del tenant.
5. Guardar el tenant.
6. Crear un usuario administrador para el nuevo tenant.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

7. Acceder a "Administradores".
8. Hacer clic en "Nuevo administrador".
9. Completar formulario con datos válidos.
10. Asociar al tenant recién creado.
11. Guardar el usuario
12. Cerrar sesión como superadministrador.
13. Iniciar sesión en el frontend de mensajería con las credenciales del nuevo administrador de mensajería.
14. Navegar por las diferentes secciones de la aplicación.

Resultado: El tenant se crea exitosamente y el administrador del nuevo tenant puede acceder a la aplicación, todas las páginas y secciones deben mostrar información vacía (sin datos), validando el correcto aislamiento de datos entre tenants y no se debe mostrar información de otros tenants existentes en el sistema.

9.1.19 CP-019: Visualización de ruta de un pedido en mapa

Prerrequisitos: Tener pedido con direcciones de recogida y entrega válidas, ser usuario autorizado.

Descripción: Consultar la ruta de un pedido específico utilizando la funcionalidad de mapa integrada.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Ejecución:

1. Acceder a lista de pedidos.
2. Seleccionar pedido con direcciones válidas.
3. Hacer clic en "Ver ruta".
4. Verificar visualización del mapa.
5. Revisar ruta sugerida.

Resultado: El mapa se carga correctamente mostrando la ubicación de recogida, entrega y la ruta óptima sugerida entre ambos puntos.

9.1.20 CP-020: Creación de cliente con pedidos anteriores

Prerrequisitos: Tener pedidos sin cliente registrado, ser usuario autorizado.

Descripción: Verificar la creación correcta de un usuario utilizando direcciones de pedidos anteriores que no contaban con la información del cliente.

Ejecución:

1. Ingresar a la sección de clientes.
2. Darle clic a "Nuevo cliente".
3. Ingresar campos correctamente.
4. Utilizar la funcionalidad de pedidos sin cliente.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

5. Seleccionar un pedido de los de la lista.
6. Guardar cliente.
7. Verificar la creación correcta con las direcciones seleccionadas.

Resultado: El mapa se carga correctamente mostrando la ubicación de recogida, entrega y la ruta óptima sugerida entre ambos puntos.

9.1.21 CP-021: Tenant inactivo

Prerrequisitos: Tener un tenant configurado en estado "inactivo" y un usuario válido asociado a dicho tenant.

Descripción: Verificar que los usuarios no pueden acceder a la aplicación cuando su tenant asociado se encuentra en estado inactivo.

Ejecución:

1. [Opcional - Como superadministrador] Verificar que el tenant esté efectivamente inactivo.
2. Iniciar sesión como superadministrador.
3. Navegar a "Tenants".
4. Localizar el tenant objetivo.
5. Confirmar que el estado sea "Inactivo".
6. Cerrar sesión.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

7. Intentar iniciar sesión con un usuario válido del tenant inactivo.
8. Ingresar credenciales correctas (usuario/email y contraseña).
9. Hacer clic en "Iniciar sesión".

Resultado: El sistema debe rechazar el acceso del usuario y mostrar un mensaje de error con credenciales inválidas con las cuales no puede acceder a la aplicación. El usuario no debe poder ingresar a ninguna funcionalidad de la aplicación de mensajería y la sesión no debe establecerse.

9.2 Ejecución de las pruebas

Las pruebas funcionales se ejecutaron utilizando el prototipo desarrollado, siguiendo cada uno de los casos de prueba definidos. Durante la ejecución se documentaron los resultados y se realizaron las correcciones necesarias hasta lograr que todos los casos fueran exitosos.

A continuación, se presentan algunas capturas de pantalla que ilustran la ejecución de casos de prueba representativos:

Caso CP-004. Creación de arqueo de caja.

Para este primer caso de prueba, se accedió al sistema como administrador de una empresa de mensajería y se ingresó a la sección de arqueos de caja. Desde allí, se hizo clic en el botón "**Nuevo arqueo**" y se completaron los datos requeridos. Se verificó que el turno seleccionado correspondiera al rango de tiempo en el que fue creado y se ingresaron los valores necesarios, siendo obligatorios únicamente el turno y el monto de efectivo inicial, ya que la fecha

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

se asigna automáticamente por el sistema. Adicionalmente, se registraron valores en los campos de efectivo real y egresos con el fin de comprobar el correcto funcionamiento del proceso de arqueo.

Figura 36

Creación arqueo de caja

The screenshot displays the 'Gestión de arqueos' interface for 'Mensajería Rápida S.A.S.'. The top navigation bar includes a search filter, a 'Mostrando 3 de 3 arqueos' indicator, a 'Reportes' button, and a '+ Nuevo arqueo' button. Below this is a search filter section with fields for 'Fecha Desde', 'Fecha Hasta', 'Turno', 'Estado', and 'Usuario'. The main content area shows the 'Nuevo arqueo' form with the following fields: 'Fecha' (07/17/2025), 'Turno' (Noche (22:00:00 - 06:00:00)), 'Efectivo Inicio' (100000), 'Efectivo Real' (200000), 'Egresos' (50000), and 'Observaciones' (Observaciones adicionales). The form has 'Guardar' and 'Cancelar' buttons at the bottom.

Una vez creado el arqueo, se verificó que el registro se haya generado correctamente. Además, se comprobó el funcionamiento del cálculo automático de la diferencia, el cual se realiza mediante la fórmula:




$$\text{Diferencia} = \text{Efectivo real} - (\text{Efectivo inicial} + \text{Total ingresos} - \text{Egresos})$$

En este caso específico, el resultado indicó una diferencia positiva de \$150.000, lo que significa que hay más efectivo del que reflejan los registros del sistema.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Figura 37

Verificación correcta creación arqueo de caja

Arqueo	Fecha	Usuario	Turno	Estado	Efectivo Inicial	Total Ingresos	Efectivo Real	Egresos	Diferencia	Acciones
#8	16/07/2025	Nombre Test Apellido Test	Noche	Abierto	\$ 100.000,00	\$ 0,00	\$ 200.000,00	\$ 50.000,00	\$ 150.000,00	  

Caso CP-014. Creación de pedido anterior de un cliente registrado.

Para este caso, lo primero que se realizó fue la verificación de la existencia de un pedido con estado "**Entregado**" en el sistema, con el objetivo de reutilizar sus datos para una creación rápida. Esto permitió obtener el nombre del cliente asociado, el cual se utilizaría posteriormente al registrar un nuevo pedido.

Figura 38

Verificación existencia pedido entregado



Gestión de pedidos Mensajería Rápida S.A.S. Mostrando 1 de 8 pedidos [+ Nuevo pedido](#)

Filtros de búsqueda Lista Tarjetas

ID pedido Estado Cliente

2 Entregado Buscar por nombre o teléfono

Desde Hasta

mm/dd/yyyy mm/dd/yyyy [Limpiar filtros](#)

ID	Cliente	Recogida	Entrega	Estado	Mensajero	Total	Fecha	Acciones
#2	Farmacia San Rafael 3109876543	Calle 56 #23-45	Carrera 45 #67-89	Entregado	Pedro Luis Sánchez Torres	\$4.850	17/07/2025	  

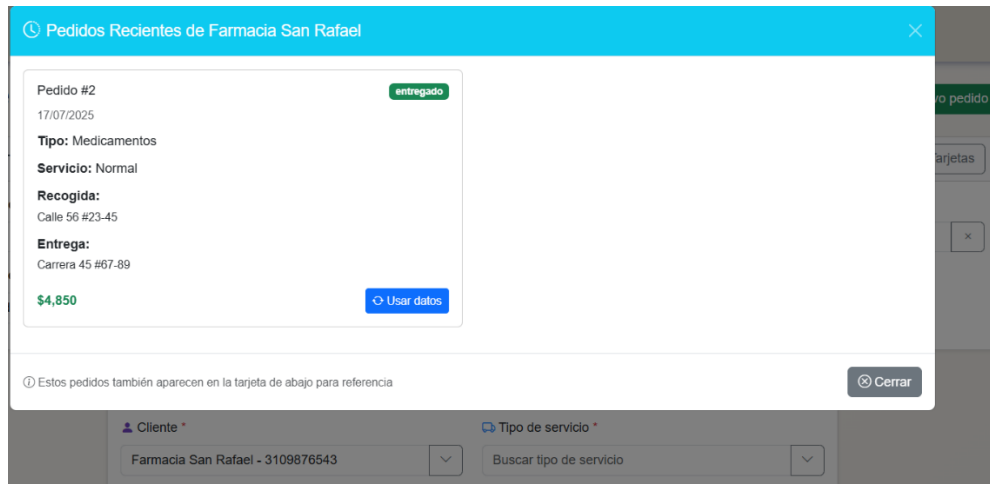
Al hacer clic en el botón "**Nuevo pedido**", se desplegó el formulario de creación. Al ingresar el nombre del cliente previamente identificado, el sistema mostró automáticamente los pedidos recientes asociados a dicho cliente, con el fin de agilizar el proceso. En caso de que el usuario cierre accidentalmente el módulo, esta misma información se encuentra disponible al

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

final del formulario. Como se desea reutilizar uno de estos registros, se debe hacer clic en el botón **"Usar datos"**.

Figura 39

Creación con funcionalidad de pedidos recientes



Inmediatamente, toda la información del pedido se carga automáticamente en el formulario, por lo que el único paso restante para el usuario es hacer clic en el botón **"Guardar"**.

Figura 40

Formulario pedido creación rápida

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Tarifa *

Normal Zona 1 - \$5000 - Entrega normal zona central

Información del paquete

Tipo de paquete * Medicamentos

Peso (kg) 0.5

Valor declarado 85000

Costo de compra 0

Subtotal Calculado automáticamente

Total Calculado automáticamente

Calculado automáticamente: Tarifa + Costo de compra

Calculado automáticamente: Subtotal - Descuento del cliente

Fecha de entrega 16/07/2025

La fecha se establece automáticamente con la fecha actual

Notas Pedido entregado exitosamente

Guardar Cancelar

Para verificar que el pedido se hubiera creado correctamente, se utilizó el filtro por nombre del cliente. De esta manera, se confirmó que el nuevo registro fue generado de forma rápida y con la misma información previamente cargada.

Figura 41

Filtro de búsqueda pedidos

Filtros de búsqueda









Lista Tarjetas

ID pedido Estado Cliente

Buscar por ID Todos los estados Farmacia

Desde Hasta

mm/dd/yyyy mm/dd/yyyy Limpiar filtros

ID	Cliente	Recogida	Entrega	Estado	Mensajero	Total	Fecha	Acciones
#10	Farmacia San Rafael 3109876543	Calle 56 #23-45	Carrera 45 #67-89	Pendiente	Sin asignar	\$4.850	17/07/2025	   
#2	Farmacia San Rafael 3109876543	Calle 56 #23-45	Carrera 45 #67-89	Entregado	Pedro Luis SÁnchez Torres	\$4.850	17/07/2025	   

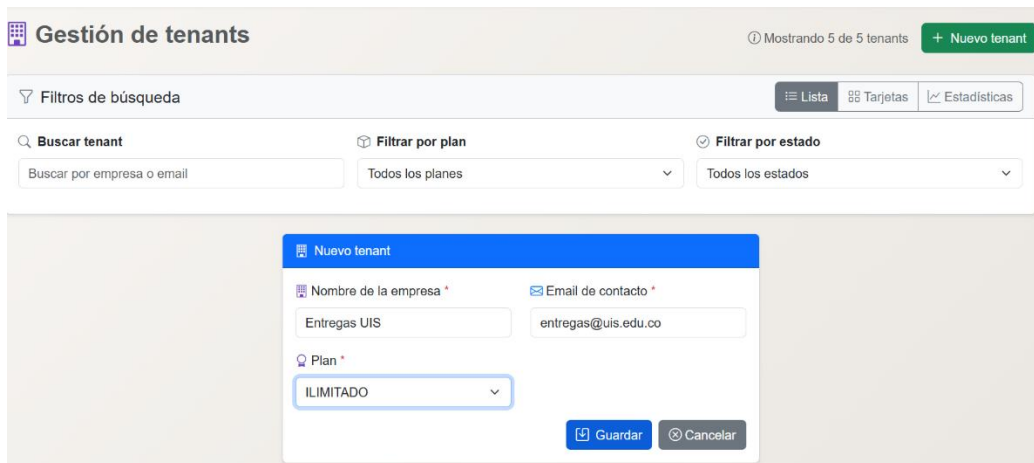
SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Caso CP-018. Creación de tenant y verificación de aislamiento de datos.

Para realizar esta prueba, lo primero que se hizo fue ingresar como superadministrador al apartado de “**Tenants**” y hacer clic en “**Nuevo tenant**”, donde se ingresó la información necesaria para su creación.

Figura 42

Creación de tenant



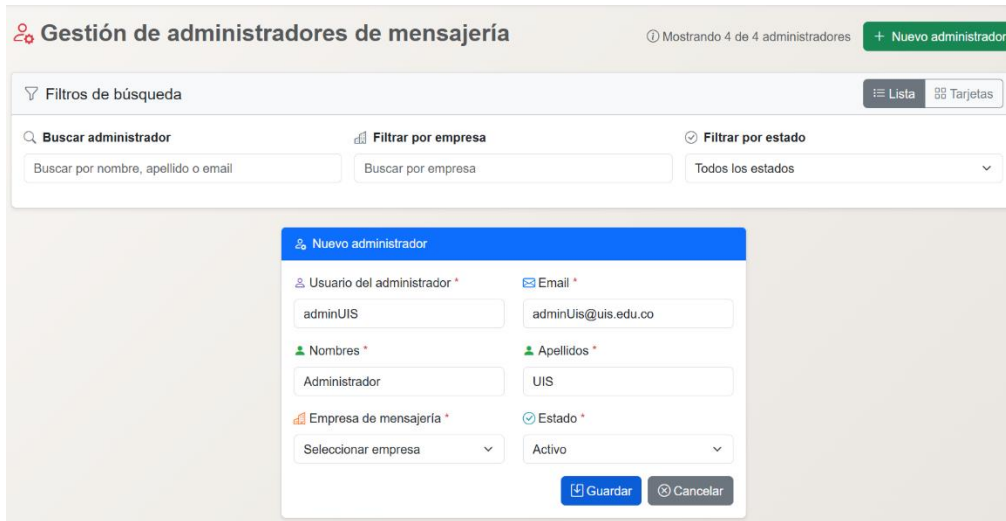
The screenshot displays the 'Gestión de tenants' (Tenant Management) interface. At the top, it shows 'Mostrando 5 de 5 tenants' and a '+ Nuevo tenant' button. Below this is a search and filter section with 'Filtros de búsqueda', 'Lista', 'Tarjetas', and 'Estadísticas' options. The search filters include 'Buscar tenant' (with a text input 'Buscar por empresa o email'), 'Filtrar por plan' (set to 'Todos los planes'), and 'Filtrar por estado' (set to 'Todos los estados'). The main content area features a modal window titled 'Nuevo tenant' with the following fields: 'Nombre de la empresa *' (filled with 'Entregas UIS'), 'Email de contacto *' (filled with 'entregas@uis.edu.co'), and 'Plan *' (a dropdown menu set to 'ILIMITADO'). At the bottom of the modal are 'Guardar' and 'Cancelar' buttons.

A continuación, se ingresó al apartado de “**Administradores**” para crear un nuevo administrador que sería asociado al tenant previamente creado. Para ello, se utilizó la opción “**Nuevo administrador**” y se ingresó la información correspondiente.

Figura 43

Creación de administrador de mensajería

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



Se verificó la correcta creación del usuario mediante la simulación del envío de correo electrónico, en el cual se recibió la contraseña temporal. Esta contraseña se utilizará posteriormente para ingresar como el administrador recién creado y comprobar el aislamiento de datos.

Figura 44

Contraseña enviada al administrador creado

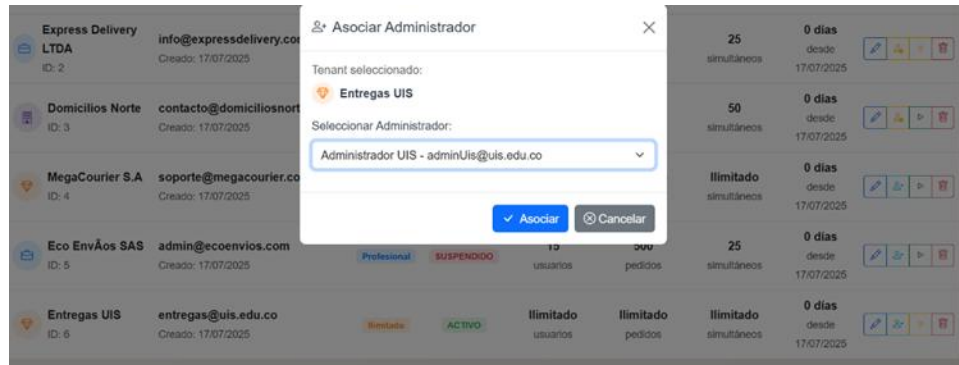
```
courier-app | Para: adminUis@uis.edu.co
courier-app | Asunto: Credenciales de acceso
courier-app | Mensaje:
courier-app | Hola adminUIS,
courier-app |
courier-app | Tu contraseña temporal es: l^0jREK+t4hr
courier-app |
```

Posteriormente, se procedió a asociar correctamente al administrador con el tenant correspondiente, permitiéndole así acceder a todas las funcionalidades asignadas.

Figura 45

Asociación administrador al tenant

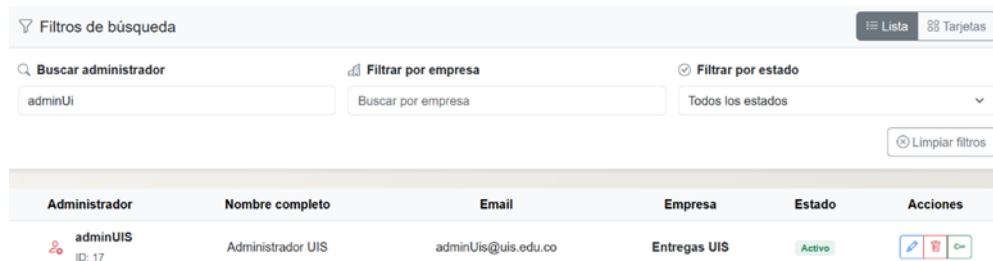
SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



Se verificó que el nuevo administrador se hubiera actualizado correctamente con el tenant al que fue asociado.

Figura 46

Verificación actualización administrador

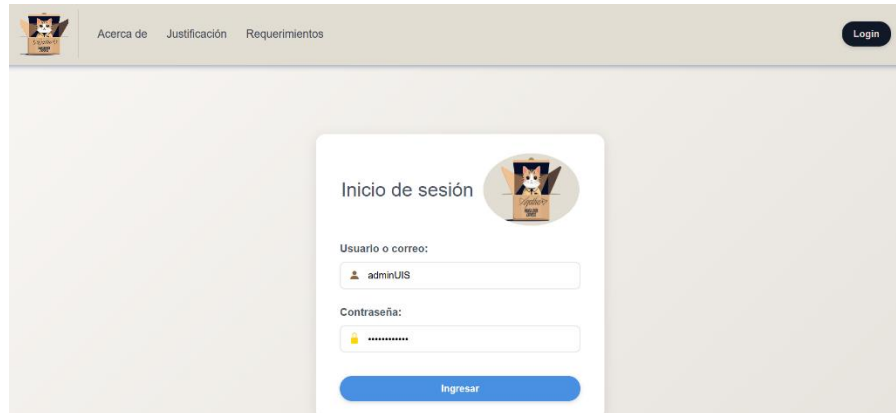


Posteriormente, se cerró sesión con el usuario superadministrador y se inició sesión con el usuario recién creado, con el fin de verificar el acceso a todas las funcionalidades.

Figura 47

Inicio de sesión con administrador creado

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



De esta manera, se pudo verificar el correcto aislamiento de datos, ya que al tratarse de un tenant nuevo, no se encontraba información registrada en ninguno de los apartados. Como prueba, se accedió a las secciones de clientes, usuarios y pedidos, donde se evidenció que el entorno pertenece a la empresa correspondiente al tenant creado al inicio de esta prueba, y que, al ser un entorno recién generado, no cuenta con registros iniciales.

Figura 48

Verificación aislamiento de datos en clientes

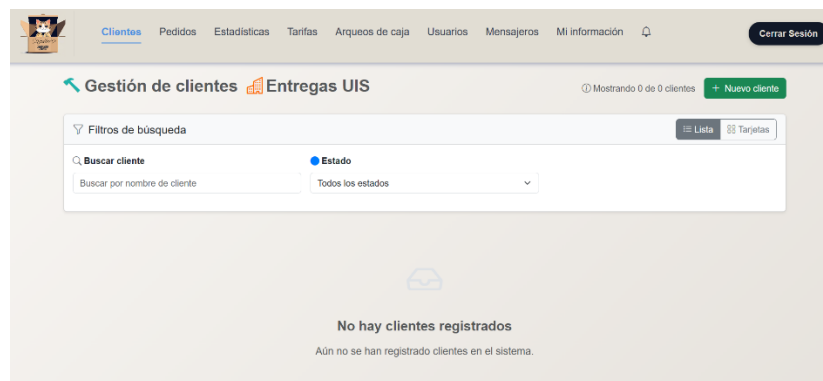


Figura 49

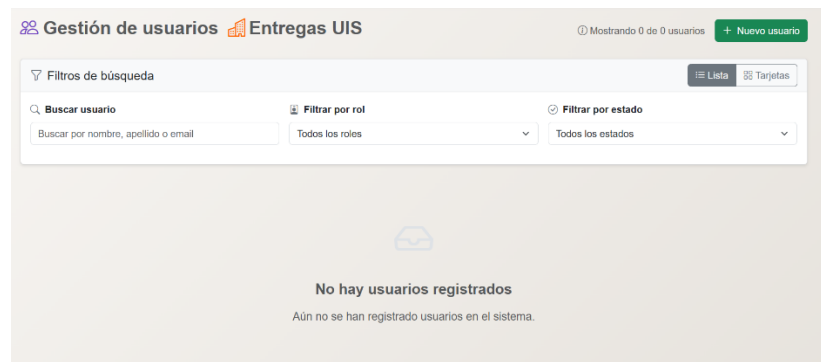
Verificación aislamiento de datos en pedidos

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



Figura 50

Verificación aislamiento de datos en usuarios



Caso CP-019. Visualización de ruta de un pedido en mapa.

Para visualizar la ruta de un pedido, es necesario que este cuente con dos direcciones válidas y reales: una de origen y otra de destino. Una vez se dispone de un pedido con estas condiciones, es posible acceder a la visualización de la ruta a seguir haciendo clic en el ícono del mapa que representa esta funcionalidad.

Figura 51





Verificación de pedido con direcciones correctas

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Filtros de búsqueda Lista Tarjetas

ID pedido: 10 Estado: Todos los estados Cliente: Buscar por nombre o teléfono

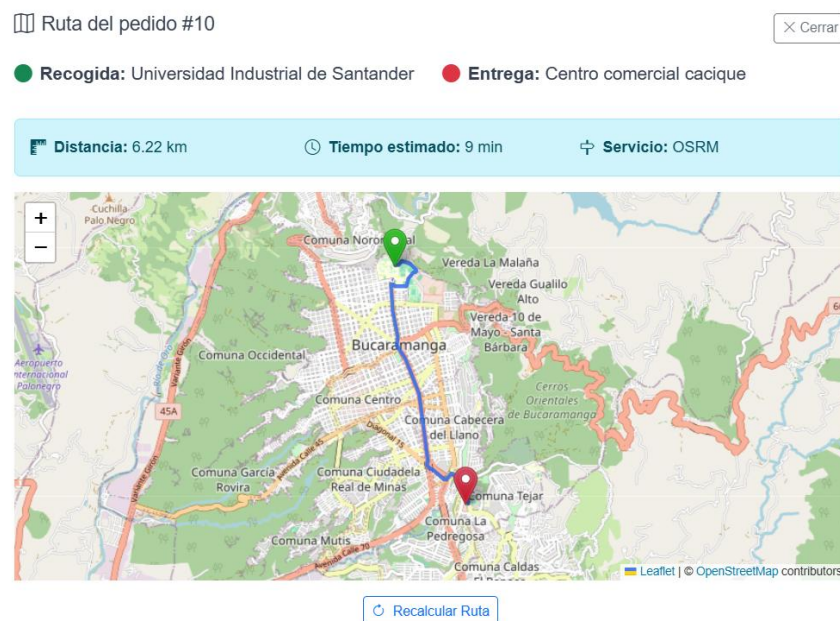
Desde: mm/dd/yyyy Hasta: mm/dd/yyyy Limpiar filtros

ID	Cliente	Recogida	Entrega	Estado	Mensajero	Total	Fecha	Acciones
#10	Farmacia San Rafael 3109876543	Universidad Industrial de Santander	Centro comercial cacique	Pendiente	Sin asignar	\$4.850	17/07/2025	   

Posteriormente, la ruta óptima se visualiza en un modal con el mapa, donde se calcula tanto la distancia como el tiempo estimado del recorrido. Esta funcionalidad utiliza **OSRM (Open Source Routing Machine)** como motor de cálculo de rutas.

Figura 52

Mapa generado por pedido



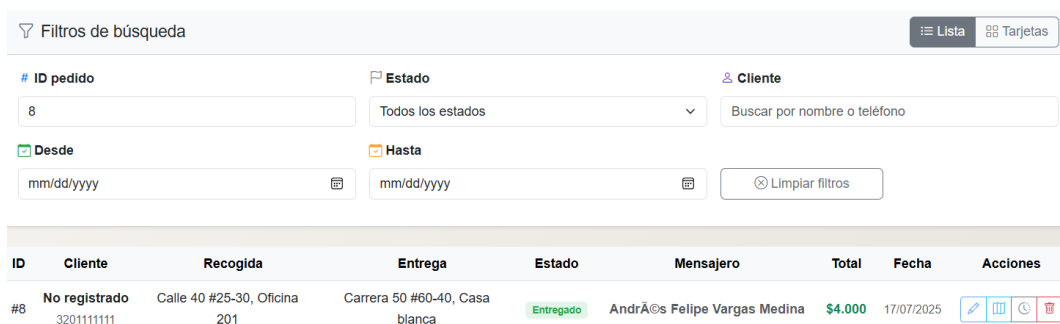
Caso CP-020. Creación de cliente con pedidos anteriores.





SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Dado que algunos usuarios del servicio de mensajería express suelen realizar pedidos de forma frecuente sin estar registrados inicialmente, se implementó una funcionalidad que permite crear clientes a partir de pedidos anteriores sin cliente asociado. Como primer paso, se verificó que efectivamente existieran pedidos en el sistema sin un cliente registrado.

Figura 53

Verificación de pedido sin cliente



ID	Cliente	Recogida	Entrega	Estado	Mensajero	Total	Fecha	Acciones
#8	No registrado 3201111111	Calle 40 #25-30, Oficina 201	Carrera 50 #60-40, Casa blanca	Entregado	Andrés Felipe Vargas Medina	\$4.000	17/07/2025	   

Contando con el ID del pedido sin cliente asociado, se accedió a la sección de clientes. Desde allí, al hacer clic en el botón "**Nuevo cliente**" e ingresar la información básica (como nombre, teléfono y descuento), es posible agilizar el proceso utilizando la funcionalidad "**Direcciones de pedido sin cliente**". Esta opción permite asociar automáticamente las direcciones del pedido identificado, facilitando así la creación rápida del nuevo cliente.

Figura 54

Formulario de creación de cliente

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Gestión de clientes Mensajería Rápida S.A.S

Mostrando 4 de 4 clientes + Nuevo cliente

Filtros de búsqueda

Lista Tarjetas

Buscar cliente Estado

Buscar por nombre de cliente Todos los estados

Nuevo cliente

Nombre * Floristeria alegría

Teléfono * 3156485203

Descuento (%) 0.00

Estado Activo

Direcciones del cliente

Direcciones pedidos sin cliente Agregar dirección

No hay direcciones agregadas. Haga clic en "Agregar dirección" para añadir una.

Guardar Cancelar

Luego de seleccionar la opción **"Direcciones de pedido sin cliente"**, se despliega un modal que muestra una lista de los pedidos sin cliente asociado. Al identificar el ID del pedido que se desea vincular, se hace clic en la opción **"Agregar direcciones"**, lo que permite asociar automáticamente las direcciones de ese pedido al nuevo cliente en proceso de creación.

Figura 55

Carga de pedidos sin cliente

Direcciones de pedidos sin cliente

Pedido #8

Recogida:
Calle 40 #25-30, Oficina 201
Bucaramanga - Centro

Entrega:
Carrera 50 #60-40, Casa blanca
Bucaramanga - Provenza

Agregar direcciones

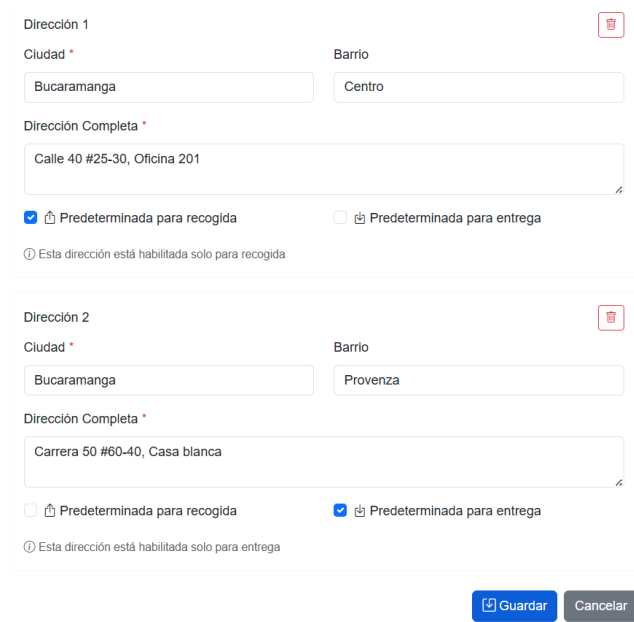
Cerrar

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Después, se verifica en el formulario que las direcciones hayan sido cargadas correctamente, asegurando así la consistencia de la información entre el pedido original y el nuevo cliente que va a ser creado, por lo que se procede a dar clic en '**Guardar**'.

Figura 56

Formulario de direcciones automatizadas



The screenshot displays a web form for managing automated addresses. It is divided into two main sections, 'Dirección 1' and 'Dirección 2', each with a delete icon in the top right corner.

Dirección 1:

- Ciudad ***: Bucaramanga
- Barrio**: Centro
- Dirección Completa ***: Calle 40 #25-30, Oficina 201
- Radio buttons: Predeterminada para recogida, Predeterminada para entrega
- Info icon: Esta dirección está habilitada solo para recogida

Dirección 2:

- Ciudad ***: Bucaramanga
- Barrio**: Provenza
- Dirección Completa ***: Carrera 50 #60-40, Casa blanca
- Radio buttons: Predeterminada para recogida, Predeterminada para entrega
- Info icon: Esta dirección está habilitada solo para entrega

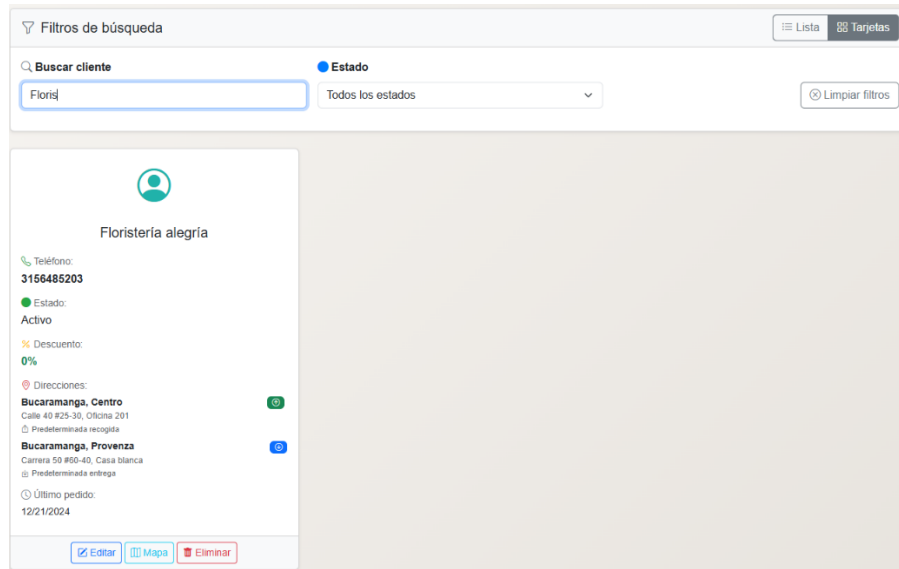
At the bottom right, there are two buttons: 'Guardar' (Save) and 'Cancelar' (Cancel).

Finalmente, se verifica que el cliente haya sido creado correctamente con las direcciones asociadas mediante la funcionalidad "**Direcciones de pedidos sin cliente**", confirmando que la información se haya registrado de forma precisa y completa.

Figura 57

Verificación creación cliente

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS



9.3 Resultados validación

La ejecución de las pruebas funcionales se llevó a cabo de manera iterativa, realizando las correcciones necesarias en cada ciclo hasta lograr que todos los casos de prueba fueran exitosos. El proceso de validación confirmó que el sistema cumple con todos los requisitos funcionales establecidos.

Tabla 2

Resultados de las pruebas de validación

Código	Caso de prueba	Resultado
CP-001	Creación de pedido	OK
CP-002	Asignación de mensajero a pedido	OK
CP-003	Registro de nuevos clientes	OK
CP-004	Creación de arqueo de caja	OK
CP-005	Edición de información mensajero	OK
CP-006	Búsqueda y filtrado de pedidos	OK
CP-007	Generación de reporte de ingresos	OK
CP-008	Actualización de estado de pedidos	OK

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

CP-009	Configuración de tarifas	OK
CP-010	Cierre de arqueo de caja	OK
CP-011	Visualización de dashboard con métricas	OK
CP-012	Gestión de notificaciones	OK
CP-013	Creación de pedido con datos inválidos	OK
CP-014	Creación de pedido anterior de un cliente registrado	OK
CP-015	Creación cuarto arqueo de caja en el mismo día	OK
CP-016	Asignación de descuento a cliente frecuente	OK
CP-017	Registro de ingreso manual en arqueo de caja	OK
CP-018	Creación de tenant y verificación de aislamiento de datos	OK
CP-019	Visualización de ruta de un pedido en mapa	OK
CP-020	Creación de cliente con pedidos anteriores	OK
CP-021	Tenant inactivo	OK

La validación confirma que el sistema de gestión de mensajería cumple satisfactoriamente con todos los requisitos funcionales establecidos, demostrando su robustez y confiabilidad para su implementación en un entorno de producción.

10. Conclusiones y trabajo futuro

Se logró diseñar exitosamente un sistema de software multitenant integral que brinda soporte robusto al negocio de mensajería express, estableciendo las bases tecnológicas para la transformación digital de pequeñas y medianas empresas del sector. El sistema propuesto demuestra capacidad para centralizar operaciones, optimizar recursos y facilitar la gestión empresarial mediante una arquitectura escalable y eficiente.

Se identificaron y documentaron las características fundamentales de las aplicaciones multitenant: aislamiento de datos, compartición de recursos, escalabilidad horizontal y gestión centralizada de la infraestructura. Este análisis permitió establecer que la arquitectura multitenant

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

ofrece una economía de escala significativa al compartir recursos computacionales entre múltiples tenants sin comprometer la seguridad ni la integridad de los datos.

Se identificaron los requerimientos críticos del sector de mensajería express: gestión de clientes, seguimiento de envíos, facturación y reportes operacionales. Este estudio evidenció la necesidad de sistemas flexibles que permitan adaptarse a diferentes modelos de negocio y escalas operativas propias de las pymes del sector.

Se diseñó una arquitectura multitenant robusta que integra componentes tales como: servicio proxy, instancia de la aplicación, servicio tenants y bases de datos compartidas, garantizando el aislamiento y la eficiencia operacional. La arquitectura propuesta implementa patrones de diseño que facilitan la escalabilidad, el mantenimiento y la evolución del sistema conforme crecen las necesidades empresariales.

Se desarrolló exitosamente un prototipo funcional que demuestra la viabilidad técnica de la arquitectura multitenant propuesta, implementando los módulos centrales del sistema de mensajería express. El prototipo confirmó la capacidad del sistema para manejar múltiples tenants de forma concurrente, manteniendo el rendimiento y la integridad de los datos.

La validación del prototipo a través de casos de uso reales demostró que el sistema aborda eficazmente las problemáticas específicas de este tipo de empresas. Se evidenció que el sistema posee la flexibilidad necesaria para adaptarse a distintos escenarios operacionales y manejar volúmenes de transacciones comunes en el sector, validando su potencial para la transformación digital del sector de mensajería express.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

La arquitectura multitenant se revela como un paradigma especialmente poderoso para las pymes, ya que modela de forma robusta la realidad empresarial donde los recursos deben ser optimizados y la escalabilidad debe ser inherente al diseño. Esta aproximación arquitectónica trasciende las limitaciones tradicionales de los sistemas monolíticos, ofreciendo una propuesta de valor que se alinea perfectamente con las necesidades y restricciones presupuestarias de las pequeñas y medianas empresas.

La implementación de una economía de escala a través de la compartición inteligente de recursos representa el aspecto más significativo de este trabajo. La 'multitenancy' no es simplemente una técnica de optimización de recursos, sino una manifestación tecnológica de principios económicos fundamentales que permiten a las pequeñas empresas acceder a capacidades que tradicionalmente estaban reservadas para organizaciones con mayor alcance económico.

El proceso de desarrollo reveló que la arquitectura multitenant actúa como un catalizador de la transformación digital, no solo por sus capacidades técnicas, sino por su capacidad para reconfigurar la relación entre las empresas y la tecnología. Al eliminar las barreras de entrada asociadas con la implementación de sistemas empresariales complejos, esta arquitectura democratiza la innovación y permite que las pymes compitan en igualdad de condiciones tecnológicas con empresas de mayor tamaño.

Las arquitecturas multitenant representan una evolución natural hacia sistemas más inclusivos, eficientes y sostenibles, capaces de adaptarse a las dinámicas empresariales contemporáneas y de escalar orgánicamente con el crecimiento de las organizaciones que las adoptan. Este trabajo confirmó que la tecnología, cuando es diseñada con visión sistémica y

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

comprensión profunda de las necesidades empresariales, puede ser un verdadero agente de transformación económica y social.

11. Recomendaciones

Con base en los resultados obtenidos y las limitaciones identificadas durante el desarrollo de este trabajo, se proponen las siguientes recomendaciones para futuras investigaciones y desarrollos:

- Diseñar arquitecturas que implementen los otros dos enfoques posibles al momento de desarrollar sistemas multi-tenant (base de datos por tenant y esquema por tenant), permitiendo realizar comparaciones de rendimiento, costos y complejidad de implementación entre las diferentes aproximaciones arquitectónicas.
- Realizar el diseño e implementación de funcionalidades innovadoras y escalables que agreguen valor al sistema, modelando nuevos procesos de negocio específicos del sector de mensajería express, tales como optimización de rutas, integración con sistemas de geolocalización en tiempo real, y automatización de procesos logísticos.
- Llevar a cabo un análisis de mercado profundo para validar la viabilidad del sistema como modelo de negocio, incluyendo estudios de factibilidad económica, análisis de competencia, estrategias de penetración de mercado y modelos de pricing para diferentes segmentos de pymes.

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

- Implementar la optimización del monitoreo de peticiones mediante la implementación de servicios de escalado automático, utilizando herramientas como Kubernetes para garantizar disponibilidad y eficiencia del sistema bajo diferentes cargas de trabajo y patrones de uso.
- Desarrollar mecanismos avanzados de seguridad y cumplimiento normativo que aborden específicamente los requerimientos del sector de mensajería express, incluyendo protección de datos personales, trazabilidad de transacciones y auditoría de operaciones.
- Realizar pruebas de rendimiento y escalabilidad en entornos de producción con cargas reales de trabajo, para validar empíricamente las capacidades del sistema y identificar posibles cuellos de botella o áreas de optimización.
- Investigar la integración del sistema con tecnologías emergentes como inteligencia artificial y machine learning para la optimización de rutas, predicción de demanda y automatización de procesos operacionales en el sector de mensajería express.

Referencias Bibliográficas

1. Cámara de Comercio de Ibagué. Base de datos de empresas y/o entidades activas - Jurisdicción Cámara de Comercio de Ibagué - Corte a 16 de julio de 2024 [Internet]. Bogotá: Datos abiertos Colombia; 2024 [citado 2025 Julio 17]. Disponible en: https://www.datos.gov.co/Comercio-Industria-y-Turismo/BASE-DE-DATOS-DE-EMPRESAS-Y-O-ENTIDADES-ACTIVAS-JU/gwqv-sqvs/about_data
2. Cámara de Comercio de Bucaramanga. Base de datos de empresas [Internet]. Bogotá: Datos abiertos Colombia; 2024 [citado 2025 Julio 17]. Disponible en: https://www.datos.gov.co/Estad-sticas-Nacionales/BASE-DE-DATOS-DE-EMPRESAS-CAMARA-DE-COMERCIO-DE-BU/wf53-j577/about_data
3. Reynoso CB. Introducción a la arquitectura de software. Buenos Aires: Universidad de Buenos Aires; 2004.
4. Software Engineering Institute. Definitions of software architecture [Internet]. Pittsburgh: Carnegie Mellon University; 2023 [citado 2025 Julio 17]. Disponible en: <https://www.sei.cmu.edu/architecture/definitions.html>
5. Garlan D, Shaw M. An introduction to software architecture (CMU-CS-94-166). Pittsburgh: Carnegie Mellon University, School of Computer Science; 1994.
6. Seethamraju R. Adoption of Software as a Service (SaaS) Enterprise Resource Planning (ERP) systems in small and medium sized enterprises (SMEs). *Information Systems Frontiers*. 2015;17:475-92. Disponible en: <http://dx.doi.org/10.1007/s10796-014-9506-5>

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

7. Benlian A, Hess T. Opportunities and risks of software-as-a-service: Findings from a survey of IT executives. *Decision Support Systems*. 2011;52(1):232-46. Disponible en: <http://dx.doi.org/10.1016/j.dss.2011.07.007>
8. Bezemer CP, Zaidman A. Multi-tenant SaaS applications: maintenance dream or nightmare? En: Proceedings of the Joint ERCIM Workshop on Software Evolution (EVOL) and International Workshop on Principles of Software Evolution (IWPSE). New York: Association for Computing Machinery; 2010. p. 88-92. Disponible en: <http://dx.doi.org/10.1145/1862372.1862393>
9. Mikowski MS, Powell JC. *Single page web applications: JavaScript end-to-end*. Greenwich: Manning Publications; 2013.
10. Scott Jr EA. *SPA design and architecture: Understanding single-page web applications*. New York: Simon and Schuster; 2015.
11. Bharadwaj A, El Sawy OA, Pavlou PA, Venkatraman N. Digital business strategy: Toward a next generation of insights. *MIS Quarterly*. 2013;37(2):471-82.
12. Venkatachalam N, Fielt E, Rosemann M, Mathews S. Small and medium enterprises using software as a service: Exploring the different roles of intermediaries. *Australasian Journal of Information Systems* [Internet]. 2014 Nov [citado 2025 Julio 17];18(3). Disponible en: <http://dx.doi.org/10.3127/ajis.v18i3.1101>
13. Chong F, Carraro G, Wolter R. Multi-tenant data architecture. MSDN Library, Microsoft Corporation. 2006:14-30.
14. Chong F, Carraro G. Architecture strategies for catching the long tail (Software-as-a-Service (SaaS)) [Internet]. Redmond: Microsoft Corporation; 2006 [citado 2025 Julio 17].

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Disponible en: <https://pamungkaswave.blogspot.com/2015/06/architecture-strategies-for-catching.html>

15. Bajenaru A. Software-as-a-Service and Cloud Computing, a solution for small and medium-sized companies. Bulletin of the Transilvania University of Brasov Series V Economic Sciences. 2010;3(52).
16. Kalra S, Prabhakar TV. Towards dynamic tenant management for microservice-based multi-tenant SaaS applications. En: Proceedings of the 11th Innovations in Software Engineering Conference. New York: Association for Computing Machinery; 2018. Article 12, p. 1-5. Disponible en: <http://dx.doi.org/10.1145/3172871.3172882>
17. Pervez Z, Lee S, Lee YK. Multi-tenant, secure, load disseminated SaaS architecture. En: Proceedings of the 12th International Conference on Advanced Communication Technology. Piscataway: IEEE Press; 2010. p. 214-9.
18. Loukis E, Janssen M, Mintchev I. Determinants of software-as-a-service benefits and impact on firm performance. Decision Support Systems. 2019;117:38-47. Disponible en: <http://dx.doi.org/10.1016/j.dss.2018.12.005>
19. Tsai W, Bai X, Huang Y. Software-as-a-service (SaaS): perspectives and challenges. Science China Information Sciences. 2014;57(1):1-15. Disponible en: <http://dx.doi.org/10.1007/s11432-013-5050-z>
20. Aulbach S, Grust T, Jacobs D, Kemper A, Rittinger J. Multi-tenant databases for software as a service: Schema-mapping techniques. En: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data. New York: Association for Computing Machinery; 2008. p. 1195-206. Disponible en: <http://dx.doi.org/10.1145/1376616.1376736>

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

21. Song H, Nguyen PH, Chauvel F, Glattetre J, Schjerpen T. Customizing Multi-Tenant SaaS by Microservices: A Reference Architecture. En: 2019 IEEE International Conference on Web Services (ICWS). Piscataway: IEEE; 2019. p. 446-8. Disponible en: <http://dx.doi.org/10.1109/ICWS.2019.00081>
22. Soldani J, Muntoni G, Neri D, Brogi A. Towards a multi-tenant microservice architecture: An industrial experience. En: 2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC). Piscataway: IEEE; 2022. p. 1026-35. Disponible en: <http://dx.doi.org/10.1109/COMPSAC54236.2022.00100>
23. Walraven S, De Borger W, Vanbrabant B, Lagaisse B, Van Landuyt D, Joosen W. Adaptive performance isolation middleware for multi-tenant SaaS. En: 2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC). Piscataway: IEEE; 2015. p. 112-21. Disponible en: <http://dx.doi.org/10.1109/UCC.2015.27>
24. Kumar A, Singh S, Sharma R. Scalable resource provisioning for multi-tenant SaaS with cloud functions. En: 2022 2nd International Conference on Cloud Computing, Performance and Quality of Service Testing (CCPQT). Piscataway: IEEE; 2022. p. 198-205. Disponible en: <http://dx.doi.org/10.1109/CCPQT56151.2022.00039>
25. Patel M, Gupta V, Arora N. Container based scalability and performance analysis of multitenant SaaS applications. En: 2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT). Piscataway: IEEE; 2022. p. 1-6. Disponible en: <http://dx.doi.org/10.1109/ICCCNT54827.2022.9984214>
26. Nguyen PH, Song H, Chauvel F, Muller R, Boyar S, Levin E. Using microservices for non-intrusive customization of multi-tenant SaaS. En: Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Foundations of Software Engineering. New York: Association for Computing Machinery; 2019. p. 905-15. Disponible en: <http://dx.doi.org/10.1145/3338906.3340452>

27. Oliveira MMA, Lima RCS, Costa MVL, Trindade CS, Queiroz PGG. SPL integrated with microservices: A hybrid architectural proposal for multitenant SaaS. En: Proceedings of the 17th Brazilian Symposium on Software Components, Architectures, and Reuse. New York: Association for Computing Machinery; 2023. p. 1-10. Disponible en: <http://dx.doi.org/10.1145/3622748.3622749>

SISTEMA MULTITENANT PARA MENSAJERÍA EXPRESS

Apéndices

Apéndice A. **Enlace al repositorio donde se encuentra todo el código trabajo.**

[GitHub - dilanerey06/multitenant-software-agatha: Agatha: SOFTWARE QUE IMPLEMENTA ARQUITECTURA MULTITENANT PARA APOYAR EL NEGOCIO DE MENSAJERÍA EXPRESS DIRIGIDO A PEQUEÑAS Y MEDIANAS EMPRESAS.](https://github.com/dilanerey06/multitenant-software-agatha)

Apéndice B. **Enlace al repositorio donde se encuentran todos los recursos digitales.**

[Trabajo de grado - Google Drive](#)

Apéndice C. **Conjunto de videos demostrativos del prototipo implementado.**

[Prototipo Agatha - YouTube](#)

Apéndice D. **Listado completo de casos de uso desarrollados.**

[Casos de prueba Agatha - YouTube](#)