



TACO API COMMERCIAL ORDERING SYSTEM (TACOS)

Documentación



CHACÓN FERNÁNDEZ ALEJANDRO
DOMÍNGEZ CARMONA JOSÉ JAVIER

DARKMODE

DarkmodeULTD@gmail.com

22/06/2023

Veracruz, México

Contenido

1. Introducción	6
1.1. Propósito.....	6
1.2. Alcance.....	6
1.2.1. Características Principales	6
1.3. Definiciones, acrónimos y abreviaciones.....	8
1.4. Referencias	9
2. Requerimientos	10
2.1. Modelo de Contexto de T.A.C.O.S.	10
2.2. Clases de Usuario	11
2.3. Requisitos del Usuario	12
2.3.1. Diagrama de actores del sistema.....	12
2.3.2. Diagrama de Paquetes.....	13
2.3.3. Diagrama de Casos de Uso	14
2.3.4. Descripciones de Casos de Uso	18
2.4. Requisitos Funcionales.....	60
2.4.1. Requisitos Funcionales Por Caso de Uso.....	60
2.4.2. Requisitos Funcionales Generales	62
2.4.3. Requisitos No Funcionales	63
2.5. Restricciones	63
3. Diseño Arquitectónico	63
3.1. Introducción	63
3.2. Definición de Requerimientos.....	64
3.3. Modelo de Procesos del Negocio	64
3.3.1. Modelo de proceso “Ordenar In Situ”	64
3.3.2. Modelo de proceso “Ordenar Remoto”	65
3.3.3. Modelo de proceso “Realizar pedido”	66
3.3.4. Modelo de proceso “Contratar empleado”	67
3.4. Modelo de Conceptos del Negocio	68
3.5. Visión del Sistema.....	69
3.6. Tabla de Verificación de Creación y Destrucción	70
3.7. Tabla de Verificación de Actualización de Relaciones.....	71
3.8. Modelo de Paquetes de Casos de Uso	72

3.8.1.	Modelo de casos de uso: Paquete “Manejo de clientes”	73
3.8.2.	Modelo de casos de uso: Paquete “Gestión de pedidos”	74
3.8.3.	Modelo de casos de uso: Paquete “Control de personal”	75
3.8.4.	Modelo de casos de uso: Paquete “Administración del Menú”	76
	Identificación de Componentes	77
3.9.	Modelo de Tipos de Negocio	77
3.10.	Modelo de Responsabilidades de Interfaces e Interfaces del Negocio.....	78
3.11.	Interfaces del Sistema.....	79
3.12.	Especificación de Componentes´.....	80
3.12.1.	Especificación de Componentes del Negocio	80
3.12.2.	Especificación de Componentes del Sistema.....	81
3.12.3.	Especificación de Componentes Externos	82
3.13.	Arquitectura de Componentes	82
4.	Construcción	83
4.1.	Pila tecnológica	83
4.1.1.	Backend (API)	83
4.1.2.	Frontend (cliente de escritorio).....	83
4.2.	Estándar de Programación.....	84
4.2.1.	Reglas de nombrado.....	84
4.2.2.	Variables	84
4.2.3.	Métodos.....	85
4.2.4.	Clases	85
4.2.5.	Interfaces.....	85
4.2.6.	Componentes gráficos.....	85
4.2.7.	Estilo del Código.....	85
4.2.8.	Comentarios	85
4.3.	Análisis Estático	87
4.4.	Buenas Prácticas de Programación Implementadas	88
4.4.1.	Nombramiento	88
4.4.2.	Comentarios	89
4.4.3.	Portabilidad	89
4.4.4.	Patrones de diseño y Reutilización	90
4.4.5.	Implementación de interfaces externas o existentes	92

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

4.4.6. Documentación del API.....	92
5. Pruebas.....	94
5.1. Casos de Prueba	94
5.1.1. Login.....	94
5.1.2. Miembro.....	97
5.1.3. Menu	103
5.1.4. Pedido	105
5.1.5. Puestos	109
5.1.6. Turnos	110
5.1.7. Reseñas	110
5.1.8. Staff.....	112
5.1.9. Cobertura total de servicios	117
ANEXO A. Reglas del negocio	118
Tabla 1. Conceptos, acrónimos y abreviaciones	8
Tabla 2. clases de usuario	11
Ilustración 1 árbol de características de T.A.C.O.S.....	7
Ilustración 2. diagrama de contexto de T.A.C.O.S.	10
Ilustración 3 Diagrama de actores t.a.c.o.s	12
Ilustración 4 Diagrama de paquetes de casos de uso T.A.C.O.S	13
Ilustración 5 diagrama de casos de uso del paquete “administración del menú”	14
Ilustración 6 diagrama de casos de uso del paquete "control de personal”	15
Ilustración 7 diagrama de casos de uso del paquete "gestión de pedidos”	16
Ilustración 8 diagrama de casos de uso del paquete "manejo de clientes”	17
Ilustración 9 Proceso del negocio “Ordenar In Situ”	64
Ilustración 10 Proceso del negocio “Ordenar Remoto”	65
Ilustración 11 Proceso del negocio “Realizar pedido”	66
Ilustración 12 Proceso del negocio “Contratar empleado”	67
Ilustración 13 Modelo de conceptos del negocio.....	68
Ilustración 14 Modelo de Paquetes de Casos de Uso "TACOS”	72
Ilustración 15 Modelo de casos de uso del paquete “Manejo de clientes”	73
Ilustración 16 Modelo de casos de uso del paquete “Gestión de pedidos”	74
Ilustración 17 Modelo de casos de uso del paquete “Manejo de personal”	75
Ilustración 18 Modelo de casos de uso del paquete “Administración del Menú”	76
Ilustración 19 Modelo de Tipos del Negocio.....	77
Ilustración 20 Diagrama de Modelo de Responsabilidad de Interfaces Inicial	78
Ilustración 21 Modelo de interfaces del sistema.....	79
Ilustración 22 Especificación de Componentes del Negocio.....	80

Ilustración 23 Especificación de Componentes del Sistema.....	81
Ilustración 24 Especificación de Componentes Externos	82
Ilustración 25 ARQUITECTURA DE componentes T.A.C.O.S.....	82

1. Introducción

1.1. Propósito

El presente documento detalla la Especificación de Requisitos de Software (ERS) para el *Taco API Commercial Ordering System* (TACOS), el cual es una interfaz de programación de aplicaciones (por sus siglas en inglés, API) que busca sistematizar el proceso, perteneciente a una taquería, para el ordenamiento de comida a domicilio, así como la integración de un sistema de ofertas y retroalimentación de los clientes de dicho establecimiento.

Este documento de ERS describe los requisitos funcionales y no funcionales para la primera versión del TACOS. Su lectura está destinada a desarrolladores y administradores del proyecto, fomentando el establecimiento de una comprensión común de los puntos planteados en el documento de *Visión y Alcance* (Chacón y Domínguez, 2023), los cuales incluyen la problemática a resolver, el valor que se espera entregar del producto de software, y las métricas de éxito que son necesarias alcanzar por el proyecto.

1.2. Alcance

1.2.1. Características Principales

A continuación, se detallan las características iniciales con las que TACOS deberá contar para la satisfacción de las necesidades del negocio.

Para el caso del presente proyecto se priorizaron los casos de uso marcados en negritas:

CAR-01: Administración del Menú

1. **Consulta de Menú**
2. Control de Alimentos
 - a. Crear Alimento
 - b. Modificar Alimento**
 - c. Eliminar Alimento
3. Control de Categorías de Alimentos
 - a. Crear Categoría de Alimentos
 - b. Modificar Categoría de Alimento
4. Control de ofertas
 - a. Publicar Oferta
 - b. Consultar Ofertas
 - c. Renovar Ofertas

CAR-02: Gestión de Pedidos

1. **Control de Pedidos**
 - a. Hacer Pedido
 - b. Ver Historial de Pedidos

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

- c. Gestionar Pedido
- 2. Gestión de Reportes**
 - a. Generación de Reporte de Pedidos del Día
 - b. Generación de Reporte de Pedidos del Mes

CAR-03: Manejo de Personas

- 1. Control de Miembros
 - a. Inscribir Miembro**
 - b. Actualizar perfil
- 2. Control de Personal
 - a. Dar Empleado de Alta**
 - b. Modificar Empleado
 - c. Consultar Empleados
- 3. Reseñas
 - a. Publicar Reseña
 - b. Consultar Reseñas**
 - c. Eliminar Reseñas**

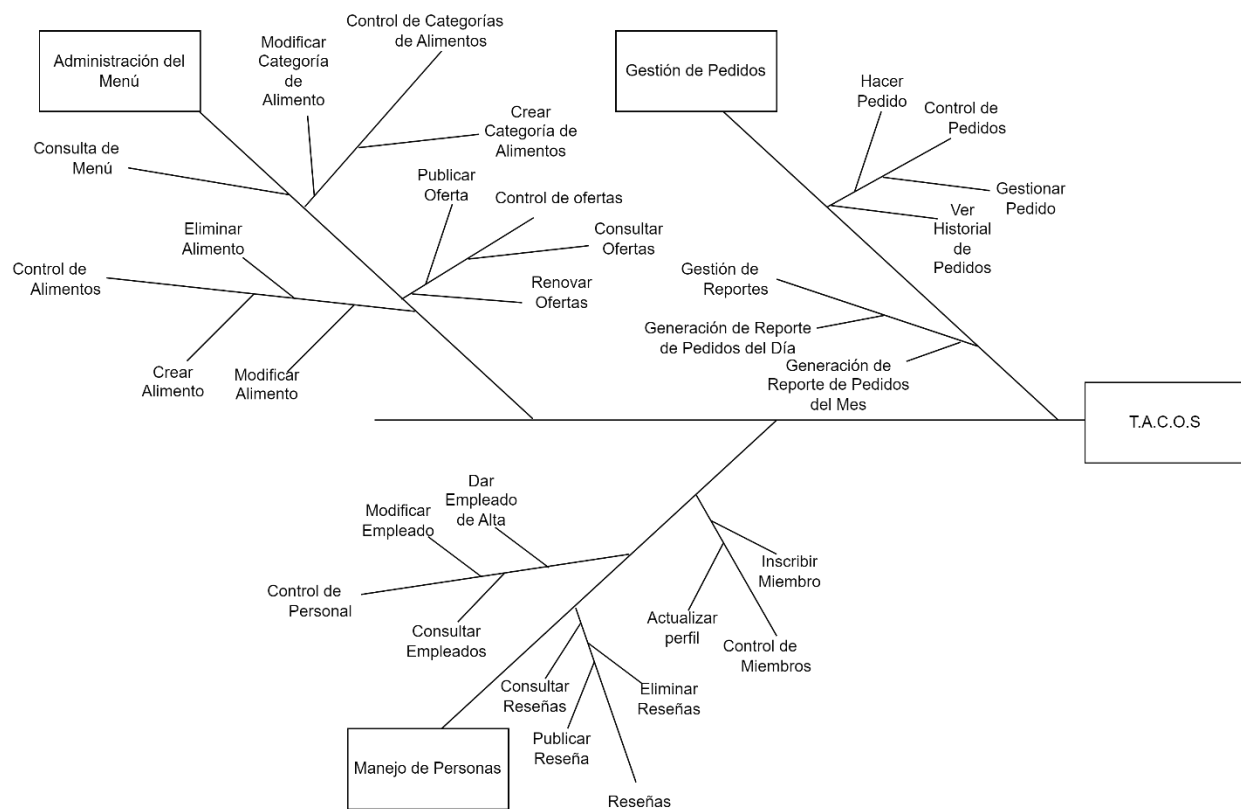


ILUSTRACIÓN 1 ÁRBOL DE CARACTERÍSTICAS DE T.A.C.O.S.

1.3. Definiciones, acrónimos y abreviaciones

En este documento se hace uso de abreviaturas que ayudan a referencias palabras de uso común dentro del contexto del negocio o que por estilo de formato mejoran la legibilidad.

Concepto	Definición
TACOS	Abreviatura del Taco API Commercial Order System el cual es el sistema que será creado.
UI	Identificador de las interfaces de usuario, donde UI proviene de las palabras en inglés User Interface.
API	Tipo de aplicación que será construida en este proyecto, que proviene de las siglas inglés de Application Programming Interface.
ME-XX	Etiquetas usadas para identificar las Métricas de Éxito donde “XX” hace referencia al número de secuencia de la etiqueta.
RF-XX	Etiquetas usadas para identificar los Requisitos Funcionales donde “XX” hace referencia al número de secuencia de la etiqueta.
RNF-XX	Etiquetas usadas para identificar los Requisitos No Funcionales donde “XX” hace referencia al número de secuencia de la etiqueta.
ACT-XX	Etiquetas usadas para identificar los Actores del sistema donde “XX” hace referencia al número de secuencia de la etiqueta.
CU-XX	Etiquetas usadas para identificar los Casos de Uso donde “XX” hace referencia al número de secuencia del caso de uso.
CAR-XX	Etiquetas usadas para identificar las Características del sistema donde “XX” hace referencia al número de secuencia de la característica.
REST-XX	Etiquetas usadas para identificar las Restricciones del sistema donde “XX” hace referencia al número de secuencia de la restricción.
REP-XX	Etiquetas usadas para identificar las Reglas de Programación del sistema, definidas en la sección 4.2, Estándar de Programación.

TABLA 1. CONCEPTOS, ACRÓNIMOS Y ABREVIACIONES

1.4. Referencias

Bass, L., Clements, P., & Kazman, R. (2021). *Software Architecture in Practice*. SEI Series in Software Engineering.

Chacón, A., Domínguez, J. J. (2023, febrero 22). T.A.C.O.S.: Taco API Commercial Ordering System – Visión y Alcance. Xalapa, Veracruz, México. Obtenido de: https://uvmx-my.sharepoint.com/:b:/g/personal/zs20015745_estudiantes_uv_mx/EWYB4VXmQJtEtCfIA93jrX8BY3YW3GZPpI4XQzLRk0i2lw?e=2DHpm7

Wiegers, K., & Beatty, J. (2013). *Software Requirements*. Pearson Education.

2. Requerimientos

2.1. Modelo de Contexto de T.A.C.O.S.

Para mejor ilustrar el alcance y delimitaciones de la presente especificación de requisitos de software, se presenta el diagrama de contexto para T.A.C.O.S.

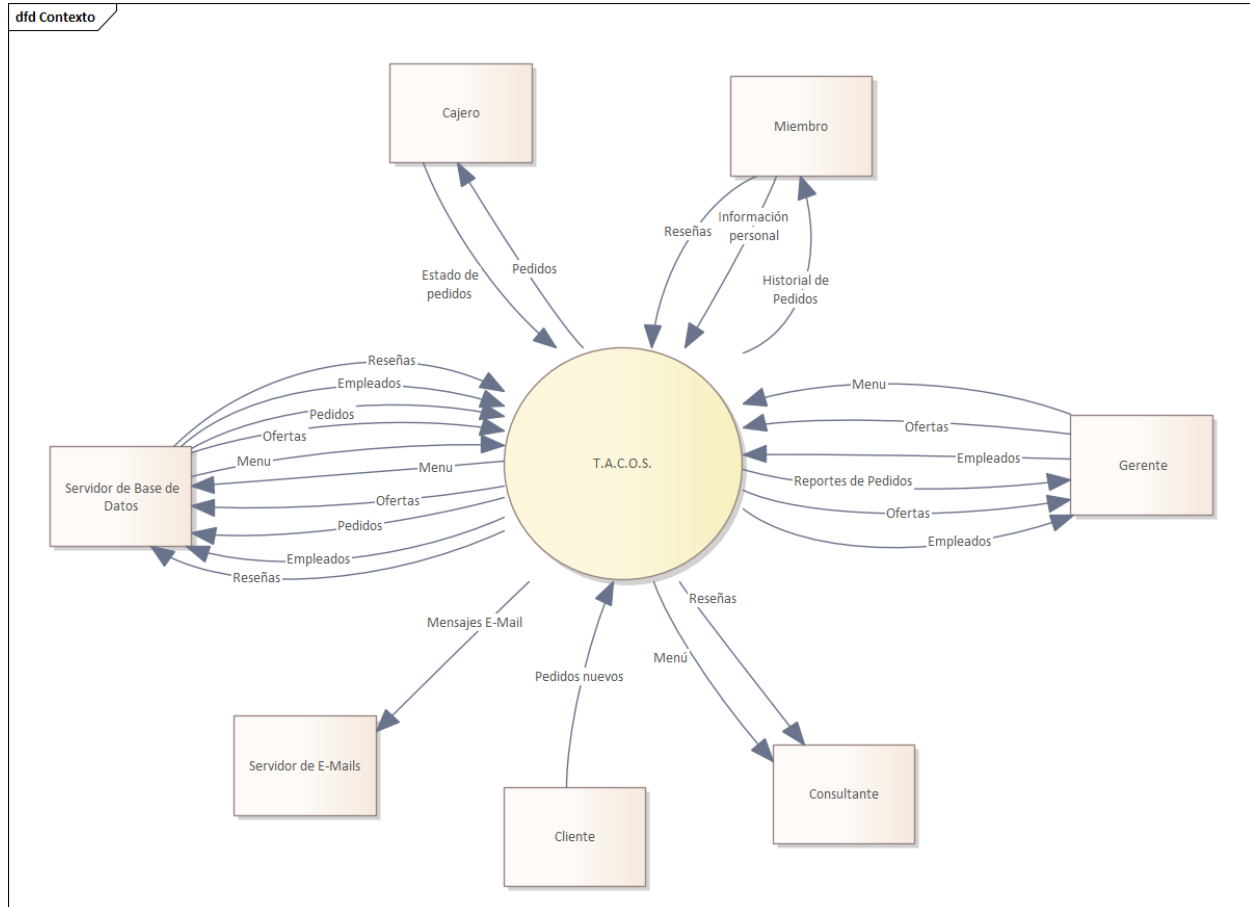


ILUSTRACIÓN 2. DIAGRAMA DE CONTEXTO DE T.A.C.O.S.

2.2. Clases de Usuario

Se han identificado los siguientes usuarios, los cuales jugarán un rol en el sistema planteado.

Rol	Tipo de usuario	Descripción
Invitado	Favorecidos	Un cliente sin una cuenta registrada en el sistema. Ellos realizarán pedidos sin tener acceso a un historial de compras ni promociones. El pago de un pedido realizado deberá ser confirmado por un <i>Cajero</i> .
Miembro	Favorecido.	Cliente con mayor funcionalidad: podrá ver su historial de pedidos y tendrá acceso a promociones según la frecuencia y volumen de sus compras.
Cajero	Favorecido.	Persona que se encarga de recibir los pedidos y registrar en el sistema si ya fueron entregados lo que da por consecuencia el registro del pago de dicho pedido.
Gerente	Ninguna.	Individuo con la responsabilidad de gestionar los inventarios del sistema, así como de generar reportes sobre los movimientos de ventas del negocio.

TABLA 2. CLASES DE USUARIO

2.3. Requisitos del Usuario

2.3.1. Diagrama de actores del sistema

En el siguiente diagrama se especifican los actores que están contemplados para el sistema, mostrando la relación de herencia que existe entre ellos.

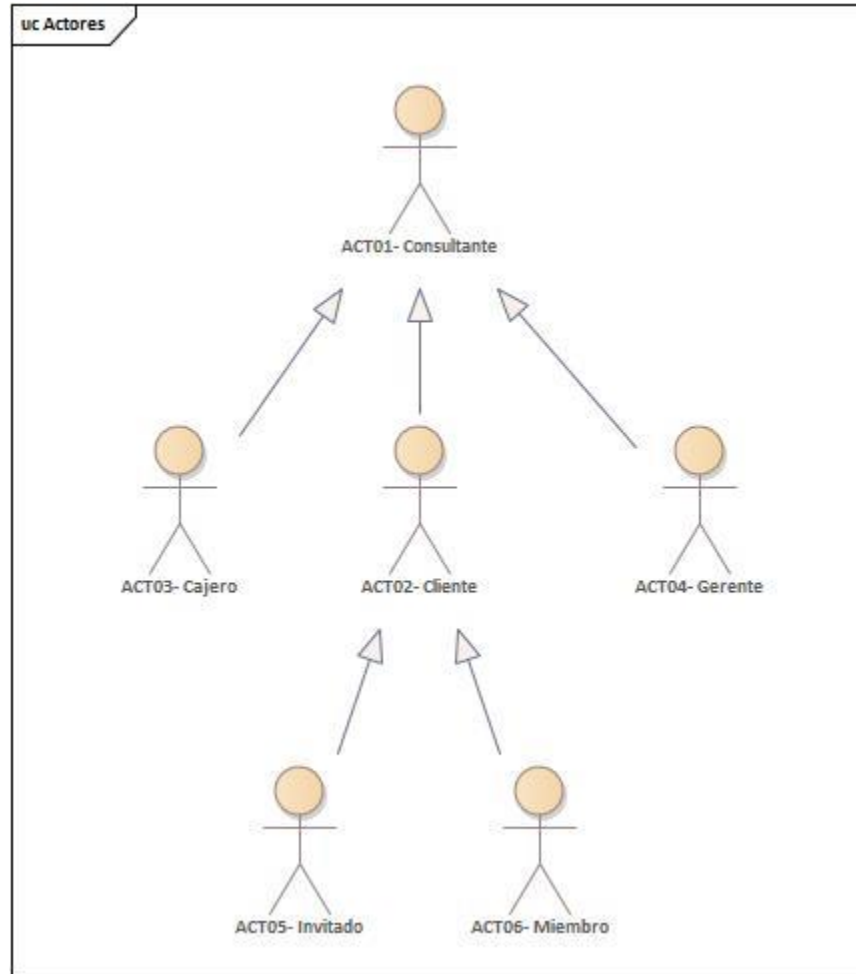


ILUSTRACIÓN 3 DIAGRAMA DE ACTORES T.A.C.O.S

2.3.2. Diagrama de Paquetes

A continuación, se define el modelo de casos de uso obtenido a partir de la agrupación de los casos de uso con base en la funcionalidad que contemplan, comenzando por el empaquetado de los casos de uso. Mediante un análisis de los casos de uso encontrados se definieron cuatro paquetes de casos de uso principales junto con el paquete de actores. Los paquetes de casos de uso identificados son: Manejo de Clientes, Control de Personal, Gestión de Pedidos, Administración del Menú y Actores.

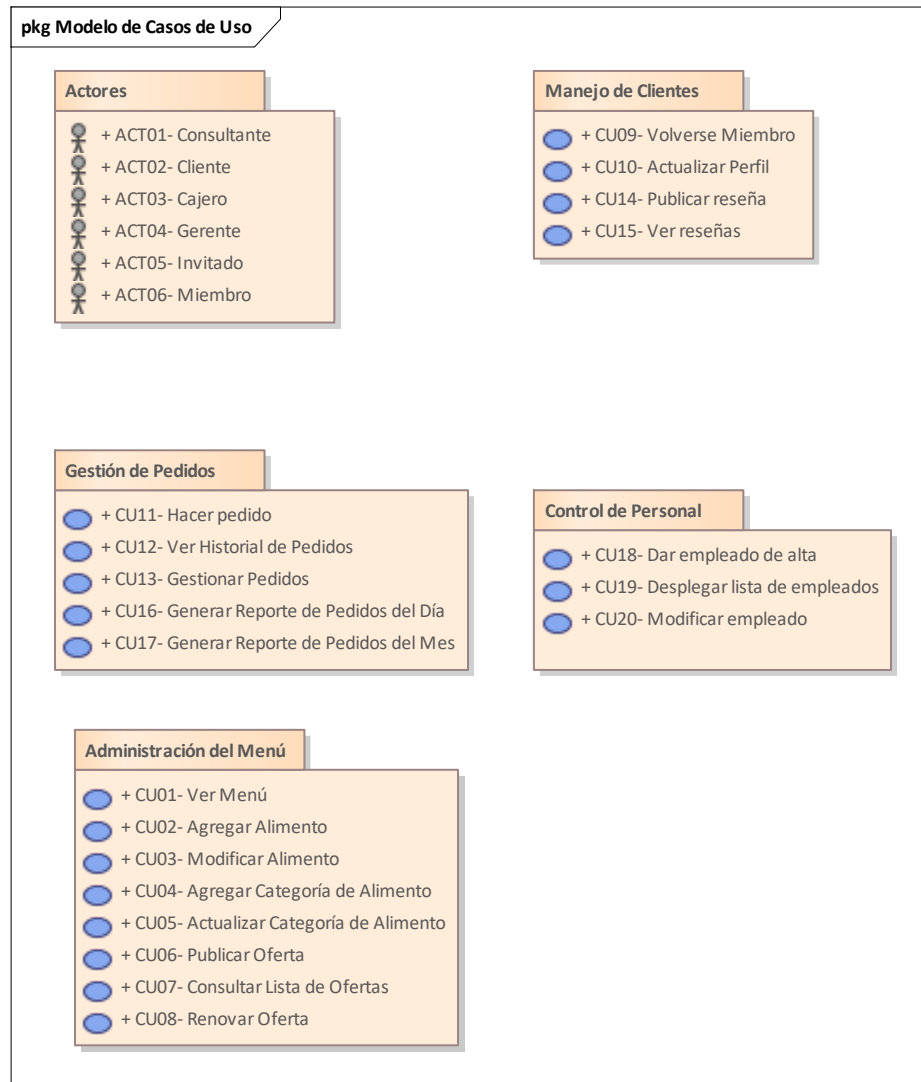


ILUSTRACIÓN 4 DIAGRAMA DE PAQUETES DE CASOS DE USO T.A.C.O.S

2.3.3. Diagrama de Casos de Uso

2.3.3.1. Diagrama de Casos de Uso: Paquete “Administración del Menú”

El siguiente diagrama muestra los casos de uso del paquete “Administración del Menú” para la gestión de alimentos y ofertas dentro del sistema.

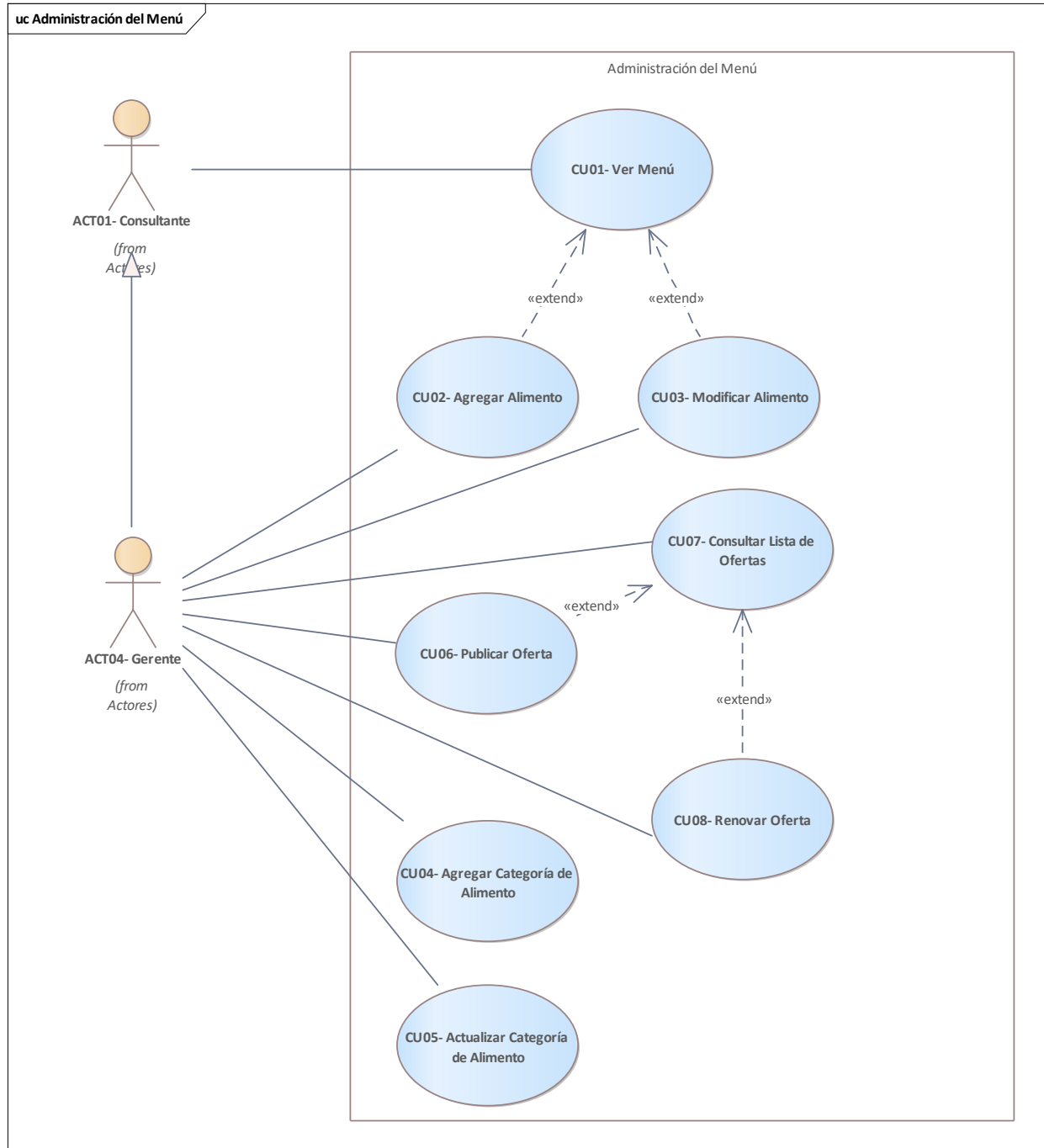


ILUSTRACIÓN 5 DIAGRAMA DE CASOS DE USO DEL PAQUETE “ADMINISTRACIÓN DEL MENÚ”

2.3.3.2. Diagrama de Casos de Uso: Paquete “Control de Personal”

El siguiente diagrama muestra los casos de uso del paquete “Control de Personal” para la gestión del personal en el sistema.

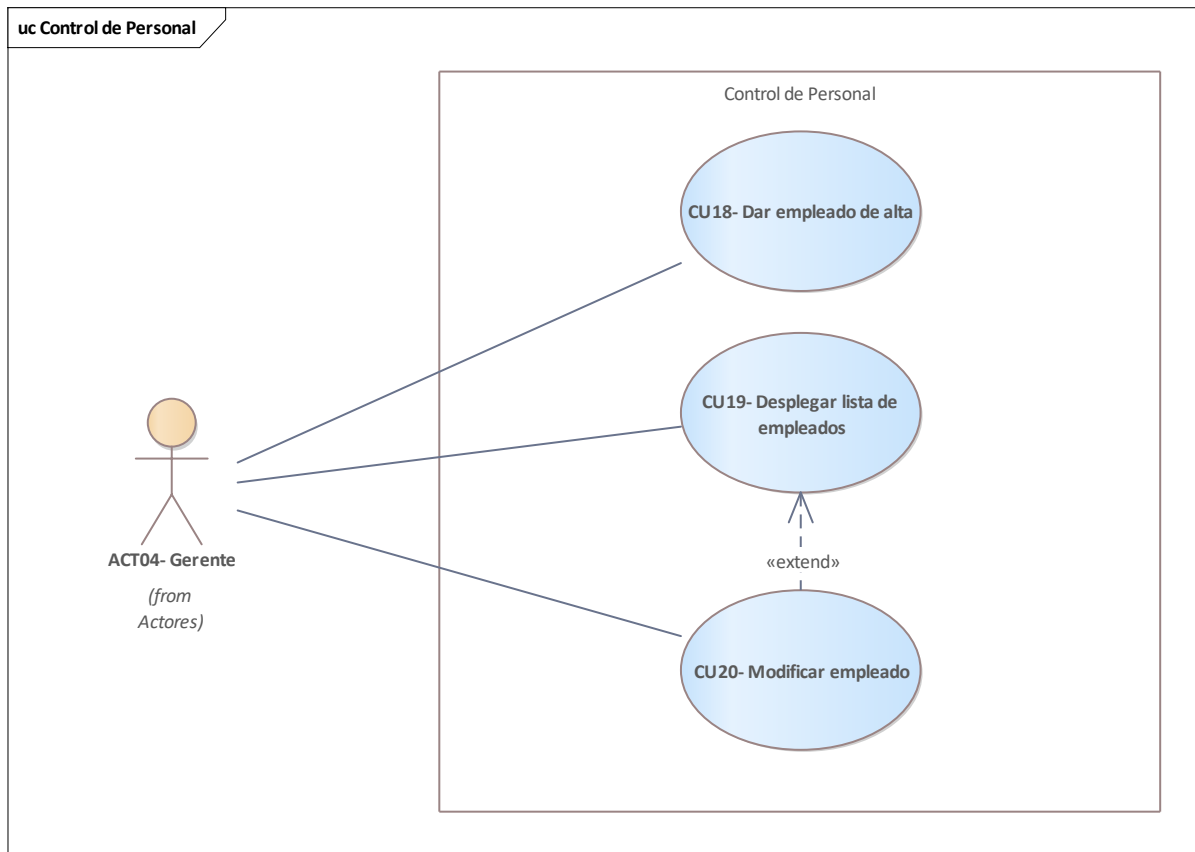


ILUSTRACIÓN 6 DIAGRAMA DE CASOS DE USO DEL PAQUETE "CONTROL DE PERSONAL"

2.3.3.3. Diagrama de Casos de Uso: Paquete “Gestión de Pedidos”

El siguiente diagrama muestra los casos de uso del paquete “Gestión de Pedidos” para la administración de los pedidos y reportes en el sistema.

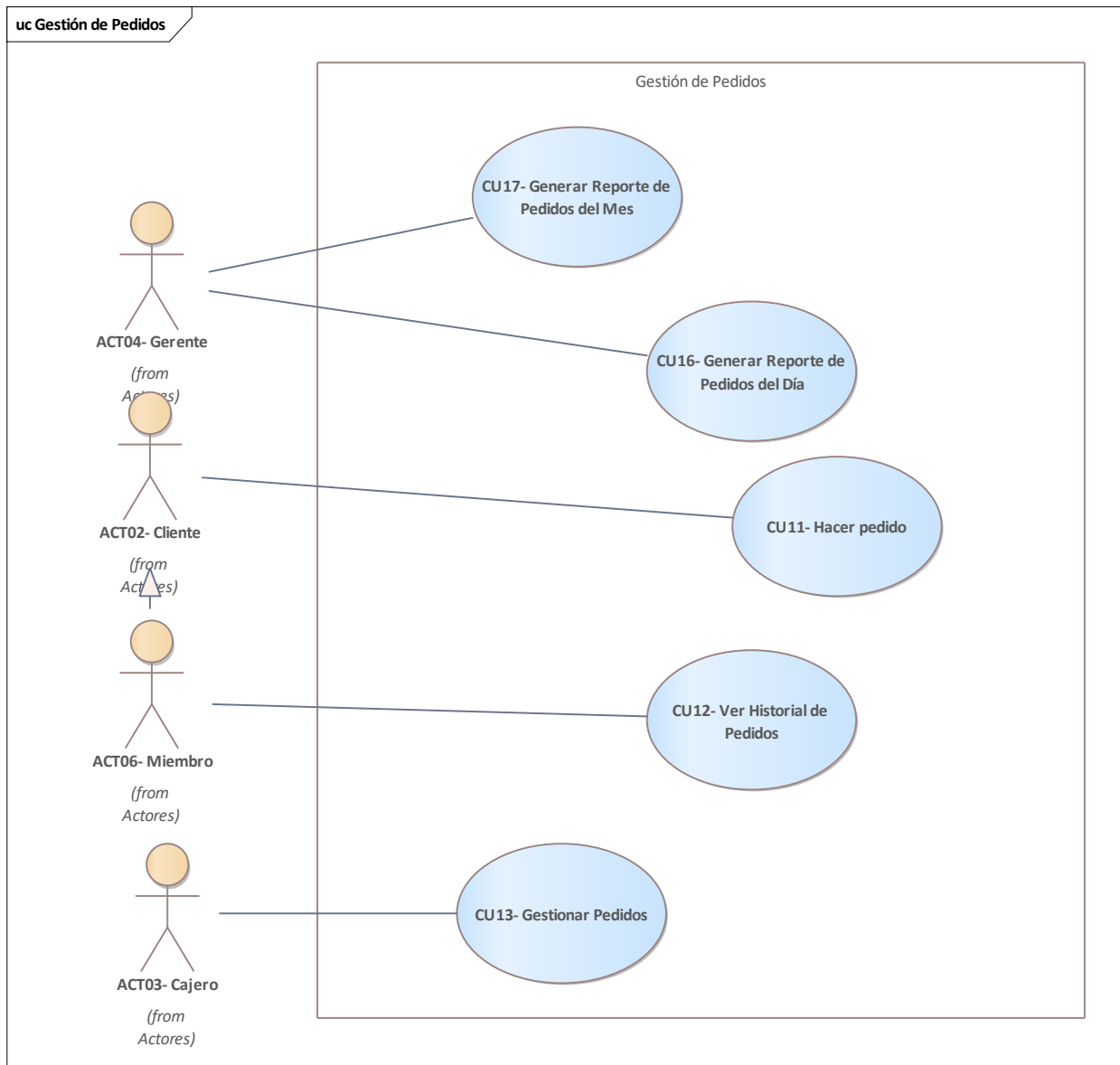


ILUSTRACIÓN 7 DIAGRAMA DE CASOS DE USO DEL PAQUETE "GESTIÓN DE PEDIDOS"

2.3.3.4. Diagrama de Casos de Uso: Paquete “Manejo de Clientes”

El siguiente diagrama muestra los casos de uso del paquete “Manejo de Clientes” para la administración las reseñas y clientes dentro del sistema.

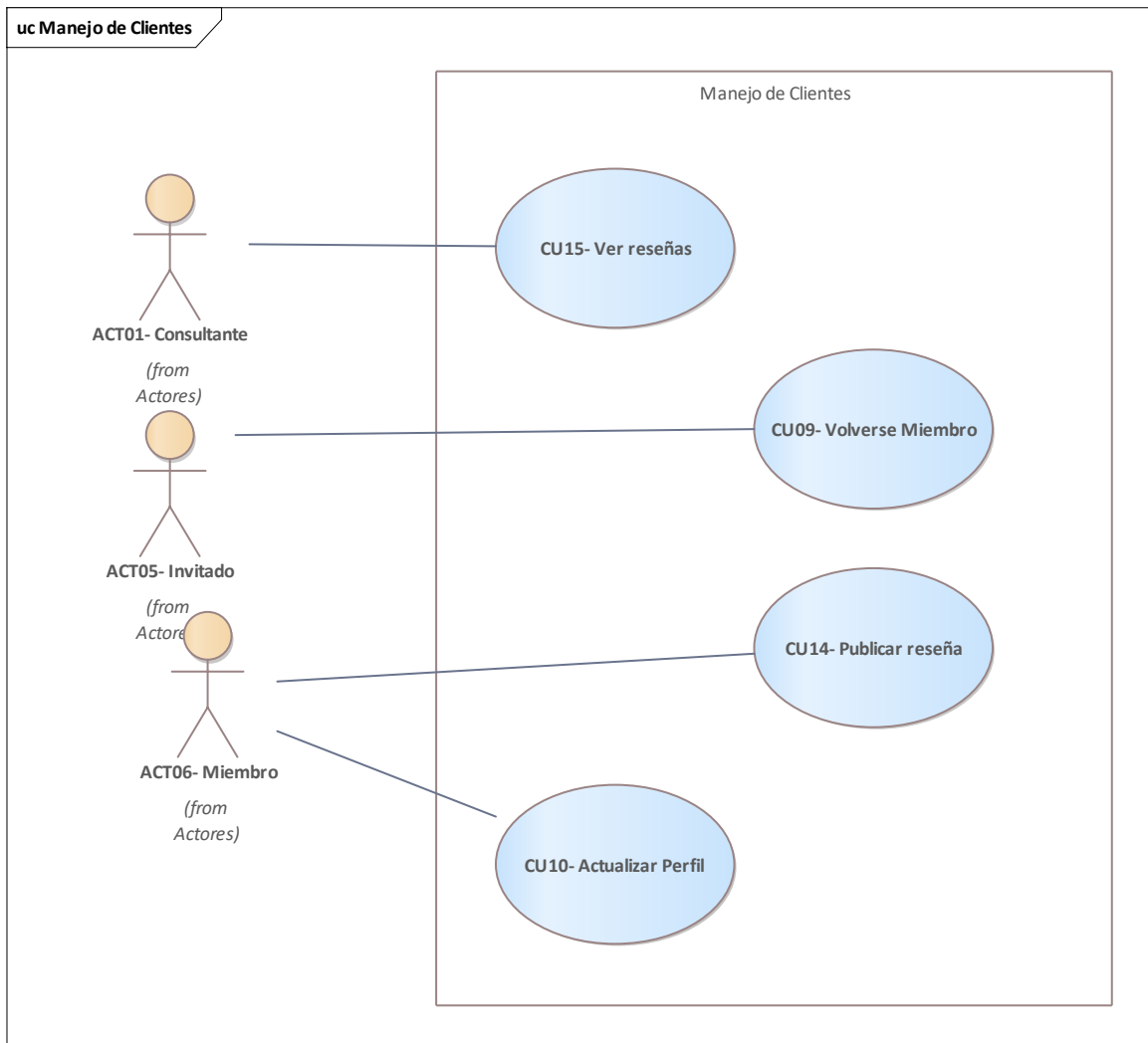


ILUSTRACIÓN 8 DIAGRAMA DE CASOS DE USO DEL PAQUETE "MANEJO DE CLIENTES"

2.3.4. Descripciones de Casos de Uso

2.3.4.1. CU-01. Ver Menú

ID	CU-01
Nombre	Ver Menú
Responsable	Alejandro Chacón
Actualización	13/03/2023
Descripción	Permite al Consultante desplegar los ALIMENTOS disponibles.
Actor	Consultante (cliente, Cajero y Gerente)
Disparador	El consultante da clic en “Menú”
Precondiciones	Ninguna.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema recupera los ALIMENTOS y CATEGORÍAS registradas en la base de datos (ver EX1). 2. El sistema muestra “GUI Menú”, con pestañas para seleccionar las distintas CATEGORÍAS recuperadas. Cada pestaña muestra una tabla para ALIMENTOS correspondientes, con columnas para el nombre, total de existencia, imagen, porción, precio y descripción de cada ALIMENTO. Cada ALIMENTO tiene el botón “Incluir”, el cual sólo está disponible si la existencia del ALIMENTO es mayor de 0. Incluye el botón “Regresar” y un campo para filtrar alimentos por nombre. 3. El sistema valida si el consultante es un Gerente (ver FA3.1;) 4. El consultante da clic en el botón “Regresar” (ver FA4.1; FA4.2; FA4.3; FA4.4; FA4.5). 5. Termina el caso de uso
Flujos alternos	<p>FA3.1. El consultante es Gerente</p> <ol style="list-style-type: none"> 1. El sistema habilita los botones “Modificar” y “Eliminar” para cada alimento en las tablas. 2. Regresa al paso 3 del flujo normal. <p>FA4.1. El cliente da clic en “Incluir” en algún ALIMENTO disponible.</p> <ol style="list-style-type: none"> 1. Extiende al CU-10. 2. Regresa al paso 4 del flujo normal. <p>FA4.2. El cliente ingresa valor en el filtro.</p> <ol style="list-style-type: none"> 1. El sistema filtra el menú para mostrar los ALIMENTOS que coincidan con el valor ingresado. 2. Regresa al paso 4 del flujo normal. <p>FA4.3. El cliente da clic en una pestaña distinta.</p> <ol style="list-style-type: none"> 1. El sistema muestra los ALIMENTOS pertenecientes a la CATEGORÍA de la pestaña seleccionada. 2. Regresa al paso 4 del flujo normal.

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

	<p>FA4.4. El Gerente da clic en “Agregar alimento”</p> <ol style="list-style-type: none"> 1. Extiende al CU-02. Agregar Alimento. 2. Regresa al paso 4 del flujo normal. <p>FA4.5. El Gerente da clic en “Modificar alimento”</p> <ol style="list-style-type: none"> 1. Extiende al CU-03. Modificar Alimento. 2. Regresa al paso 4 del flujo normal.
Excepciones	<p>EX1. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde” y el botón “Aceptar.” 2. El consultante da clic en el botón “Aceptar” 3. Termina el caso de uso.
Postcondiciones	POST-1. El consultante logró consultar el menú.
Reglas de negocio	RN-01, RN-02.
Incluye	Ninguno.
Extiende	<p>CU-02. Agregar Alimento.</p> <p>CU-03. Modificar Alimento.</p> <p>CU-10. Hacer Pedido.</p>

2.3.4.2. CU-02. Agregar Alimento

ID	CU-02
Nombre	Agregar alimento
Responsable	Javier Domínguez
Actualización	14/03/2023
Descripción	Permite al Gerente agregar un nuevo ALIMENTO.
Actor	Gerente.
Disparador	El Gerente da clic en “Agregar alimento”
Precondiciones	PRE-1. Hay al menos una CATEGORÍA registrada en la base de datos.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema recupera del sistema las CATEGORÍAS. 2. El sistema abre “GUI Alimento” y muestra los campos para introducir el nombre, CATEGORÍA, existencia, imagen, porción, precio y descripción del nuevo ALIMENTO. Muestra los botones “Agregar” y “Cancelar”. 3. El Gerente ingresa la información del nuevo ALIMENTO y da clic en “Agregar” (ver FA3.1). 4. El sistema valida que los campos estén llenos. 5. El sistema valida que los campos de texto (nombre, descripción) tengan entradas alfanuméricas. (ver FA5.1) 6. El sistema valida que los campos numéricos (existencia, porción, precio) tengan entradas numéricas (ver FA6.1) 7. El sistema valida que la imagen pese 1 MB o menos (ver FA7.1). 8. El sistema guarda el nuevo alimento en la base de datos y muestra el mensaje “operación exitosa”. (ver FA8.1) 9. Termina el caso de uso.
Flujos alternos	<p>FA3.1 El Gerente da clic en “Cancelar”.</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Cancelar operación” con el mensaje “Esta seguro de cancelar” y los botones “Sí” y “No”. 2. El Gerente da clic en “Sí”. 3. Termina el caso de uso. <p>FA5.1. El Gerente deja campos vacíos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Campos vacíos detectados.” 2. Regresa al paso 3 del flujo normal <p>FA6.1. El Gerente ingresa caracteres especiales en campo alfanumérico.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Sólo se permiten letras y números en el campo: [nombre de campo].” 2. Regresa al paso 3 del flujo normal. <p>FA7.1. El Gerente ingresa caracteres no numéricos en campo numérico.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Sólo se permiten números en: [nombre de campo].” 2. Regresa al paso 3 del flujo normal. <p>FA8.1. El Gerente ingresa una imagen con un peso superior a 1 MB.</p>

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

	<ol style="list-style-type: none"> 1. El sistema muestra el mensaje “No puede subir imágenes con un peso superior a un (1) MB” 2. Regresa al paso 3 del flujo normal.
Excepciones	<p>EX1. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde” y el botón “Aceptar.” 2. El Gerente da clic en el botón “Aceptar” 3. Termina el caso de uso.
Postcondiciones	POST-1. El Gerente registró un nuevo ALIMENTO en la base de datos.
Reglas de negocio	RN-02, RN-06.
Incluye	Ninguno.
Extiende	Ninguno.

2.3.4.3.CU-03. Modificar Alimento

ID	CU-03
Nombre	Modificar alimento.
Responsable	Alejandro Chacón
Actualización	14/03/2023
Descripción	Permite al Gerente modificar un nuevo ALIMENTO.
Actor	Gerente.
Disparador	El Gerente da clic en “Agregar alimento”
Precondiciones	PRE-1. Debe existir por lo menos un alimento registrado en el sistema. PRE-2. Debe existir al menos una categoría registrada en el sistema.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema recupera las CATEGORÍAS previamente obtenidas y habilita los campos de la “GUI Menú”: “Nombre”, “Categoría”, “Total de existencias”, “Imagen”, “Porción”, “Precio” y “Descripción” y muestra los botones “Cancelar” y “Guardar”. 2. El Gerente ingresa los datos del ALIMENTO que desea modificar y da clic en “Guardar”. 3. El sistema valida que todos los campos de ALIMENTO estén llenos. (Ver FA3.1). 4. El sistema valida que los campos “Nombre” y “Descripción” sean de tipo alfanuméricos. (Ver FA4.1) 5. El sistema valida que los campos “Total de existencias”, “Porción” y “Precio” sean numéricos. (Ver FA5.1) 6. El sistema guarda en la base de datos los datos modificados del ALIMENTO. (Ver EX01, FA6.1). 7. El sistema muestra el mensaje “Operación exitosa”. 8. Termina el caso de uso.
Flujos alternos	<p>FA3.1 El Gerente da clic en “Cancelar”.</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Cancelar operación” con el mensaje “Esta seguro de cancelar” y los botones “Sí” y “No”. 2. El Gerente da clic en “Sí”. 3. Termina el caso de uso. <p>FA4.1 Campos vacíos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Campos vacíos detectados”. 2. Regresa al paso 2del flujo normal. <p>FA5.1 Campos “Nombre” y “Descripción” no alfanuméricos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “Solo se permiten letras y números en: Nombre y Descripción”. 2. Regresa al paso 2 del flujo normal. <p>FA6.1 Campos “Total de existencias”, “Porción” y “Precio” no numéricos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Solo se permiten números en: Total de existencias, Porción y Precio”.

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

Excepciones	<p>EX01. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde” y el botón “Aceptar.” 2. El Gerente da clic en el botón “Aceptar” 3. Termina el caso de uso.
Postcondiciones	<p>POST-1. El Gerente pudo modificar los datos del alimento de manera exitosa.</p> <p>POST-2. Los nuevos datos del ALIMENTO se registraron en el sistema exitosamente.</p>
Reglas de negocio	RN-02.
Incluye	Ninguna.
Extiende	Ninguna.

2.3.4.4.CU-04. Agregar Categoría de Alimento

ID	CU-04
Nombre	Agregar Categoría de Alimento.
Responsable	Javier Domínguez
Actualización	16/03/2023
Descripción	Permite al Gerente agregar una nueva CATEGORÍA al alimento.
Actor	Gerente.
Disparador	El Gerente da clic en “Nueva categoría”.
Precondiciones	PRE-1. Hay al menos un GERENTE registrado en el sistema.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Nueva categoría” con los campos de texto para ingresar el “Nombre” de la CATEGORÍA y la “Porción”, además muestra una lista desplegable para el “Tipo de unidad”. 2. El Gerente ingresa los datos de la nueva CATEGORÍA y selecciona el “Tipo de unidad”. 3. El sistema valida que todos los campos estén llenos. 4. El sistema valida que los campos de “Nombre” y “Descripción” sean de tipo alfanumérico. (Ver FA4.1) 5. El sistema valida que el nombre de la CATEGORÍA no este registrado en la base de datos. (Ver FA5.1) 6. El sistema guarda la CATEGORÍA en la base de datos. (Ver FA6.1, EX01) 7. El sistema muestra el mensaje “Operación exitosa”. 8. El sistema muestra los campos de texto en vacío. 9. Termina el caso de uso.
Flujos alternos	<p>FA4.1. Campos vacíos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Campos vacíos detectados”. 2. Regresa al paso del flujo normal. <p>FA5.1 Campos “Nombre” y “Descripción” de tipo no alfanumérico.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Solo se permiten letras y números en: los campos Nombre y Descripción”. 2. Regresa al paso 2 del flujo normal. <p>FA6.1. CATEGORÍA existente en la base de datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Categoría existente” con el mensaje “La categoría ya se encuentra registrada” y el botón “Aceptar”. 2. El Gerente da clic en el botón “Aceptar”. 3. Regresa al paso 2 del flujo normal.
Excepciones	<p>EX01. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde” y el botón “Aceptar.” 2. El Gerente da clic en el botón “Aceptar”

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

	3. Termina el caso de uso.
Postcondiciones	POST-1. El Gerente pudo agregar una nueva CATEGORÍA de manera exitosa.
Reglas de negocio	RN-02.
Incluye	Ninguna.
Extiende	Ninguna.

2.3.4.5. CU-05. Actualizar Categoría de Alimento

ID	CU-05
Nombre	Actualizar Categoría de Alimento.
Responsable	Alejandro Chacón
Actualización	16/03/2023
Descripción	Permite al Gerente agregar un nueva CATEGORÍA al alimento.
Actor	Gerente.
Disparador	El Gerente da clic en el botón “Modificar”.
Precondiciones	PRE01-Existe por lo menos una CATEGORÍA de ALIMENTO registrada en la base de datos.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema habilita los campos para modificar el “Nombre” de la CATEGORÍA, la “Porción” y el “Tipo de unidad” que se encuentran en el panel “Detalles de la categoría”. 2. El Gerente ingresa los datos de la nueva CATEGORÍA y selecciona el “Tipo de unidad”. 3. El sistema valida que todos los campos estén llenos. 4. El sistema valida que los campos de “Nombre” y “Descripción” sean de tipo alfanumérico. (Ver FA4.1) 5. El sistema valida que el nombre de la CATEGORÍA no este registrado en la base de datos. (Ver FA5.1) 6. El sistema guarda la CATEGORÍA en la base de datos. (Ver FA6.1, EX01) 7. El sistema muestra el mensaje “Operación exitosa”. 8. El sistema muestra los campos de texto en vacío. 9. Termina el caso de uso.
Flujos alternos	<p>FA4.1. Campos vacíos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Campos vacíos detectados”. 2. Regresa al paso del flujo normal. <p>FA5.1 Campos “Nombre” y “Descripción” de tipo no alfanumérico.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Solo se permiten letras y números en: los campos Nombre y Descripción”. 2. Regresa al paso 2 del flujo normal. <p>FA6.1. CATEGORÍA existente en la base de datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Categoría existente” con el mensaje “La categoría ya se encuentra registrada” y el botón “Aceptar”. 2. El Gerente da clic en el botón “Aceptar”. 3. Regresa al paso 2 del flujo normal.
Excepciones	<p>EX01. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde” y el botón “Aceptar.” 2. El Gerente da clic en el botón “Aceptar”

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

	3. Termina el caso de uso.
Postcondiciones	POST-1. Los nuevos datos de la CATEGORÍA se registraron correctamente en la base de datos.
Reglas de negocio	RN-02.
Incluye	Ninguna.
Extiende	Ninguna.

2.3.4.6. CU-06. Publicar Oferta

ID	CU-06
Nombre	Publicar Oferta
Responsable	Javier Domínguez
Actualización	21/03/2023
Descripción	Permite al Gerente agregar un nuevo ALIMENTO.
Actor	Gerente.
Disparador	El Gerente da clic en “Publicar oferta”
Precondiciones	PRE-1. Hay al menos un ALIMENTO registrado en el sistema. PRE-2. Hay al menos un MIEMBRO con 20 pedidos o más en el sistema.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema recupera de la base de datos todos los ALIMENTOS (ver EX1) 2. El sistema abre “GUI Publicar oferta” y muestra una tabla con los ALIMENTOS recuperados, con una columna para el consumo mensual de cada uno. Incluye los botones “Publicar” y “Cancelar”. 3. El Gerente selecciona uno o múltiples ALIMENTOS; ingresa un porcentaje de descuento; selecciona una fecha de inicio y una fecha de término para la nueva OFERTA; ingresa su contraseña y da clic en “Publicar oferta”. 4. El sistema valida que se seleccionó al menos un (1) ALIMENTO (ver FA4.1) 5. El sistema valida que los campos estén llenos. (ver FA5.1) 6. El sistema valida que los campos numéricos (porcentaje de descuento) tengan entradas numéricas (ver FA6.1) 7. El sistema valida que el porcentaje de descuento sea un número entero entre 1 y 100, incluyente (ver FA7.1). 8. El sistema valida que la fecha de inicio suceda antes que la fecha de término (ver FA8.1). 9. El sistema valida la contraseña en la base de datos. (ver FA9.1, EX1) 10. El sistema envía un código de confirmación al e-mail del Gerente (ver FA10.1, EX2). 11. El sistema abre “GUI Confirmación” con el mensaje “Se notificará vía e-mail a todos los MIEMBROS consentidos (con 20 o más pedidos), informándoles de esta oferta. Por favor, introduzca el código de verificación que fue enviado a su e-mail:” con un campo de texto para ingresar el código, con los botones “Confirmar” y “Cancelar”. 12. El Gerente ingresa el código de confirmación y da clic en “Confirmar” (ver FA12.1). 13. El sistema valida el código de confirmación en la base de datos. (ver EX1) 14. El sistema guarda la nueva OFERTA en la base de datos (ver FA14.1, EX1). 15. El sistema envía los datos de la OFERTA a todos los MIEMBROS con 20 pedidos o más vía e-mail (ver EX3). 16. El sistema muestra el mensaje “Operación exitosa”.

	17. Termina el caso de uso.
Flujos alternos	<p>FA4.1. El Gerente da clic en “Cancelar”.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “¿Está seguro de que quiere cancelar la operación?” con los botones “Sí” y “No”. 2. El Gerente da clic en “Sí”. 3. Termina el caso de uso. <p>FA5.1. El Gerente no selecciona un alimento.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Debe seleccionar al menos un alimento”. 2. Regresa al paso 3 del flujo normal. <p>FA6.1. El Gerente deja al menos un (1) campo vacío.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Todos los campos son obligatorios”. 2. Regresa al paso 3 del flujo normal. <p>FA7.1. El Gerente ingresa caracteres no numéricos en campo numérico.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Solo se permiten números en: porcentaje de descuento”. 2. Regresa al paso 3 del flujo normal. <p>FA8.1. El Gerente ingresa un porcentaje inválido.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “El porcentaje de descuento debe ser un número entero entre 1 y 100”. 2. Regresa al paso 3 del flujo normal. <p>FA9.1. El Gerente ingresa una fecha de término menor a la fecha de inicio.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “La fecha de término de la oferta debe suceder después de la fecha de inicio”. 2. Regresa al paso 3 del flujo normal. <p>FA10.1. El Gerente ingresa una contraseña incorrecta.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “La contraseña es incorrecta.” 2. Regresa al paso 3 del flujo normal <p>FA12.1. El Gerente da clic en “Cancelar”</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “¿Está seguro de que quiere cancelar la operación?” con los botones “Sí” y “No”. 2. El Gerente da clic en “Sí”. 3. Termina el caso de uso. <p>FA14.1. El Gerente ingresa un código inválido.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “El código es incorrecto”. 2. Regresa al paso 11 del flujo normal.

Excepciones	<p>EX1. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde” y el botón “Aceptar.” 2. El Gerente da clic en el botón “Aceptar” 3. Termina el caso de uso. <p>EX2. No hay conexión con el servidor de correos electrónicos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “No se puede enviar correos electrónicos en este momento”. 2. Termina el caso de uso. <p>EX3. No hay conexión con el servidor de correos electrónicos después de registrar una OFERTA en la base de datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “No se puede enviar correos electrónicos en este momento”. 2. El sistema elimina la OFERTA que se acaba de registrar en la base de datos. 3. Termina el caso de uso.
Postcondiciones	<p>POST-1. El Gerente registró una nueva OFERTA en la base de datos.</p> <p>POST-2. El Gerente recibió un e-mail con el código de confirmación.</p> <p>POST-3. Los miembros con 20 o más pedidos recibieron un e-mail con los datos de la oferta.</p>
Reglas de negocio	RN-03, RN-04, RN-09.
Incluye	Ninguno.
Extiende	Ninguno.

2.3.4.7. CU-07. Consultar Lista de Ofertas

ID	CU-07
Nombre	Publicar Oferta
Responsable	Alejandro Chacón
Actualización	21/03/2023
Descripción	Permite al Gerente visualizar las ofertas registradas en el sistema.
Actor	Gerente.
Disparador	El Gerente da clic en la pestaña “Ofertas”
Precondiciones	PRE-1. El Gerente autenticó sus credenciales.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema recupera de la base de datos todas las OFERTAS registradas (ver EX1) 2. El sistema muestra “GUI Ofertas”, donde aparece un listado con las OFERTAS recuperadas, mostrando sus datos: el porcentaje de descuento, la cantidad de alimentos participantes y las fechas de inicio y de fin. Cada OFERTA tiene el botón “Renovar Oferta”. Incluye el botones “Publicar oferta”. 3. El Gerente da clic en otra pestaña. 4. Termina el caso de uso.
Flujos alternos	<p>FA4.1. El Gerente da clic en “Publicar Oferta”.</p> <ol style="list-style-type: none"> 1. Extiende al CU-06. Publicar Oferta. 2. Regresa al paso 3 del flujo normal. <p>FA5.1. El Gerente da clic en “Renovar Oferta”.</p> <ol style="list-style-type: none"> 1. Extiende al CU-08. Renovar Oferta. 2. Regresa al paso 3 del flujo normal.
Excepciones	<p>EX1. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde” y el botón “Aceptar.” 2. El Gerente da clic en el botón “Aceptar” 3. Termina el caso de uso.
Postcondiciones	POST-1. El Gerente consultó las OFERTAS existentes.
Reglas de negocio	RN-03, RN-04, RN-09.
Incluye	Ninguno.
Extiende	<p>CU-06. Publicar Oferta</p> <p>CU-08. Renovar Oferta.</p>

2.3.4.8. CU-08. Renovar Oferta

ID	CU-08
Nombre	Renovar Oferta
Responsable	Javier Domínguez
Actualización	16/03/2023
Descripción	Permite al Gerente renovar una OFERTA existente.
Actor	Gerente.
Disparador	El Gerente da clic en “Historial de ofertas”.
Precondiciones	PRE-1. Hay al menos una OFERTA registrada en el sistema. PRE-2. La OFERTA seleccionada se encuentra expirada.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema habilita los campos para modificar la “Fecha inicio”, “Fecha fin”, “Porcentaje de descuento”, “Alimentos a los que aplica” que se encuentran en el panel “Detalles de la oferta” y los botones “Renovar” y “Cancelar”. 2. El Gerente modifica los campos de la OFERTA, ingresa su contraseña y da clic en “Renovar”. 3. El sistema valida que se seleccionó al menos un (1) ALIMENTO (ver FA3.1) 4. El sistema valida que los campos estén llenos. (ver FA4.1) 5. El sistema valida que los campos numéricos (porcentaje de descuento) tengan entradas numéricas (ver FA5.1) 6. El sistema valida que el porcentaje de descuento sea un número entero entre 1 y 100, incluyente (ver FA6.1). 7. El sistema valida que la fecha de inicio suceda antes que la fecha de término (ver FA7.1). 8. El sistema valida la contraseña en la base de datos. (ver FA8.1, EX1) 9. El sistema envía un código de confirmación al e-mail del Gerente (ver FA9.1, EX2). 10. El sistema abre “GUI Confirmación” con el mensaje “Se notificará vía e-mail a todos los miembros consentidos, informándoles de esta oferta. Por favor, introduzca el código de verificación que fue enviado a su e-mail:” con un campo de texto para ingresar el código, con los botones “Confirmar” y “Cancelar”. 11. El Gerente ingresa el código de confirmación y da clic en “Confirmar” (ver FA11.1). 12. El sistema valida el código de confirmación en la base de datos. (ver EX1) 13. El sistema guarda la nueva OFERTA en la base de datos (ver FA13.1, EX1). 14. El sistema envía los datos de la OFERTA a todos los MIEMBROS con estado de “Consentido” vía e-mail (ver EX3). 15. El sistema muestra el mensaje “Operación exitosa”.

Flujos alternos	<p>FA3.1. El Gerente da clic en “Cancelar”.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “¿Está seguro de que quiere cancelar la operación?” con los botones “Sí” y “No”. 2. El Gerente da clic en “Sí”. 3. Termina el caso de uso. <p>FA4.1. El Gerente no selecciona un alimento.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Debe seleccionar al menos un alimento”. 2. Regresa al paso 2 el flujo normal. <p>FA5.1. El sistema detectada campos vacíos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Todos los campos son obligatorios”. 2. Regresa al paso 2 del flujo normal. <p>FA6.1. El Gerente ingresa caracteres no numéricos en campo numérico.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Solo se permiten números en: porcentaje de descuento”. 2. Regresa al paso 2 del flujo normal. <p>FA7.1. El Gerente ingresa un porcentaje inválido.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “El porcentaje de descuento debe ser un número entero entre 1 y 100”. 2. Regresa al paso 2 del flujo normal. <p>FA8.1. El Gerente ingresa una fecha de término menor a la fecha de inicio.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “La fecha de término de la oferta debe sucedes después de la fecha de inicio”. 2. Regresa al paso 2 del flujo normal. <p>FA9.1. El Gerente ingresa una contraseña incorrecta.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “La contraseña es incorrecta.” 2. Regresa al paso 2 del flujo normal <p>FA11.1. El Gerente da clic en “Cancelar”</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “¿Está seguro de que quiere cancelar la operación?” con los botones “Sí” y “No”. 2. El Gerente da clic en “Sí”. 3. Termina el caso de uso. <p>FA13.1. El Gerente ingresa un código inválido.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “El código es incorrecto”. 2. Regresa al paso 11 del flujo normal.
-----------------	---

Excepciones	<p>EX1. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde” y el botón “Aceptar.” 2. El Gerente da clic en el botón “Aceptar” 3. Termina el caso de uso. <p>EX2. No hay conexión con el servidor de correos electrónicos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “No se puede enviar correos electrónicos en este momento”. 2. Termina el caso de uso. <p>EX3. No hay conexión con el servidor de correos electrónicos después de registrar una OFERTA en la base de datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “No se puede enviar correos electrónicos en este momento”. 2. El sistema elimina la OFERTA que se acaba de registrar en la base de datos. 3. Termina el caso de uso.
Postcondiciones	<p>POST-1. El Gerente renovó una OFERTA de manera exitosa.</p> <p>POST-2. El Gerente recibió un e-mail con el código de confirmación.</p> <p>POST-3. Los miembros con 20 o más pedidos recibieron un e-mail con los datos de la oferta.</p>
Reglas de negocio	RN-03, RN-04, RN-09.
Incluye	Ninguno.
Extiende	Ninguno.

2.3.4.9. CU-09. Volverse Miembro

ID	CU-09
Nombre	Volverse Miembro
Responsable	Alejandro Chacón
Actualización	21/03/2023
Descripción	Permite al invitado registrarse como miembro.
Actor	Invitado.
Disparador	El invitado da clic en “Unirse”.
Precondiciones	Ninguna.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema muestra “GUI Unirse” con un formulario para ingresar su nombre, apellido paterno, e-mail, dirección y contraseña. Incluye los botones “Unirse” y “Regresar” 2. El invitado ingresa los datos y da clic en “Unirse”. 3. El sistema valida que todos los campos estén llenos. (ver FA3.1) 4. El sistema valida que los campos alfabéticos (nombre y apellido paterno) tengan sólo valores alfabéticos. (ver FA4.1) 5. El sistema valida que los campos alfanuméricos (número exterior, calle y colonia) tengan sólo valores alfanuméricos (ver FA5.1). 6. El sistema valida que el teléfono tenga un formato válido (ver FA6.1). 7. El sistema valida que el e-mail tenga un formato válido (ver FA7.1). 8. El sistema valida que el e-mail no exista en la base de datos. (ver FA8.1) 9. El sistema valida que la contraseña tenga un formato válido (ver FA9.1). 10. El sistema envía un código de confirmación al e-mail proporcionado (ver FA10.1, EX2). 11. El sistema muestra “GUI Confirmación” con un campo de texto para ingresar el código enviado, y los botones “Confirmar” y “Cancelar”. 12. El invitado ingresa el código de confirmación y da clic en “Confirmar”. 13. El sistema valida el código de confirmación en la base de datos (ver FA13.1). 14. El sistema registra el nuevo MIEMBRO en la base de datos (ver FA14.1, EX1). 15. Termina el caso de uso.
Flujos alternos	<p>FA4.1. El invitado no llenó todos los campos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Campos vacíos detectados”. 2. Regresa al paso del flujo normal. <p>FA5.1 El invitado ingresó valores no alfabéticos en algún campo alfabéticos (nombre y/o apellido paterno).</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Solo se permiten letras y números en: los campos Nombre y Descripción”. 2. Regresa al paso 2 del flujo normal.

	<p>FA6.1. El invitado ingresó valores no alfanuméricos en algún campo alfanumérico (número exterior, calle y/o colonia).</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “Solo se permiten letras y números en: [Nombre del campo]”. 2. Regresa al paso 2 del flujo normal. <p>FA7.1. El invitado ingresó un teléfono con un formato erróneo.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “El campo [nombre del campo] es inválido. Debe seguir el formato: [formato]”. 2. Regresa al paso 2 del flujo normal. <p>FA8.1. El invitado ingresó un e-mail que ya fue registrado.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “Ya hay una cuenta con ese e-mail. Por favor, intenta con otro.” 2. Regresa al paso 2 del flujo normal. <p>FA9.1. El invitado ingresó un e-mail con un formato erróneo.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “El campo [nombre del campo] es inválido. Debe seguir el formato: [formato]”. 2. Regresa al paso 2 del flujo normal. <p>FA10.1. El invitado ingresó una contraseña con un formato erróneo.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “El campo [nombre del campo] es inválido. Debe seguir el formato: [formato]”. 2. Regresa al paso 2 del flujo normal. <p>FA13.1. El invitado da clic en “Cancelar.”</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “¿Está seguro de que desea cancelar?” con los botones “Sí” y “No”. 2. El invitado da clic en “Sí”. 3. Termina el caso de uso. <p>FA14.1. El invitado ingresó un código incorrecto.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “El código es incorrecto. Por favor, inténtalo de nuevo.” 2. Regresa al paso 12 del flujo normal.
Excepciones	<p>EX1. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje ““No te podemos registrar en este momento. Por favor, inténtalo más tarde” y el botón “Aceptar.” 2. El invitado da clic en el botón “Aceptar” 3. Termina el caso de uso. <p>EX2. No hay conexión con el servidor de correos electrónicos.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “No te podemos registrar en este momento. Por favor, inténtalo más tarde” y el botón “Aceptar”

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

	2. El invitado da clic en el botón “Aceptar.” 3. Termina el caso de uso.
Postcondiciones	POST-1. Se ha registrado un nuevo MIEMBRO en la base de datos.
Reglas de negocio	RN-04, RN-05.
Incluye	Ninguna.
Extiende	Ninguna.

2.3.4.10. CU-10. Actualizar Perfil

ID	CU-10
Nombre	Actualizar Perfil
Responsable	Javier Domínguez
Actualización	23/03/2023
Descripción	Permite al miembro modificar su información.
Actor	Miembro.
Disparador	El miembro da clic en “Actualizar perfil.”
Precondiciones	PRE-01. La información del MIEMBRO ya fue recuperada de la base de datos durante el inició sesión.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema muestra “GUI Actualizar Perfil” con un formulario con los campos obligatorios, llenos de la información del MIEMBRO: nombre, apellido paterno, e-mail, teléfono, número exterior, calle, colonia y descripción de domicilio. Muestra dos campos de contraseña vacíos; uno para actualizar la contraseña, y otro para confirmar la actualización con la contraseña actual. Incluye los botones “Guardar cambios”. 2. El invitado modifica los datos que requiera, ingresa su contraseña actual y da clic en “Guardar cambios”. 3. El sistema valida la contraseña actual en la base de datos (ver EX1.) 4. El sistema valida que todos los campos obligatorios estén llenos. (ver FA4.1) 5. El sistema valida que los campos alfabéticos (nombre y apellido paterno) tengan sólo valores alfabéticos. (ver FA5.1) 6. El sistema valida que los campos alfanuméricos (número exterior, calle y colonia) tengan sólo valores alfanuméricos (ver FA6.1). 7. El sistema valida que el teléfono tenga un formato válido (ver FA7.1). 8. El sistema valida que el e-mail tenga un formato válido (ver FA8.1). 9. El sistema valida que el e-mail no pertenezca a otro MIEMBRO en la base de datos (ver FA9.1). 10. El sistema valida que la contraseña tenga un formato válido, si es que el campo para actualizarla está lleno (ver FA10.1). 11. El sistema actualiza la información del MIEMBRO en la base de datos (ver FA11.1, EX1). 12. Termina el caso de uso.
Flujos alternos	<p>FA4.1. El invitado ingresó una contraseña actual errónea.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Campos vacíos detectados”. 2. Regresa al paso del flujo normal. <p>FA5.1. El invitado no llenó todos los campos obligatorios.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Campos vacíos detectados”. 2. Regresa al paso del flujo normal.

	<p>FA6.1 El invitado ingresó valores no alfabéticos en algún campo alfabéticos (nombre y/o apellido paterno).</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Solo se permiten letras y números en: los campos Nombre y Descripción”. 2. Regresa al paso 2 del flujo normal. <p>FA7.1. El invitado ingresó valores no alfanuméricos en algún campo alfanumérico (número exterior, calle y/o colonia).</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “Solo se permiten letras y números en: [Nombre del campo]”. 2. Regresa al paso 2 del flujo normal. <p>FA8.1. El invitado ingresó un teléfono con un formato erróneo.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “El campo [nombre del campo] es inválido. Debe seguir el formato: [formato]”. 2. Regresa al paso 2 del flujo normal. <p>FA9.1. El invitado ingresó un e-mail con un formato erróneo.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “El campo [nombre del campo] es inválido. Debe seguir el formato: [formato]”. 2. Regresa al paso 2 del flujo normal. <p>FA10.1. El invitado ingresó un e-mail que ya fue registrado.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “Ya hay una cuenta con ese e-mail. Por favor, intenta con otro.” 2. Regresa al paso 2 del flujo normal. <p>FA11.1. El invitado ingresó una contraseña con un formato erróneo.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “El campo [nombre del campo] es inválido. Debe seguir el formato: [formato]”. 2. Regresa al paso 2 del flujo normal.
Excepciones	<p>EX1. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje ““hay conexión con la base de datos. Por favor, inténtalo más tarde” y el botón “Aceptar.” 2. El invitado da clic en el botón “Aceptar” 3. Termina el caso de uso.
Postcondiciones	POST-1. Se ha actualizado la información de un MIEMBRO en la base de datos.
Reglas de negocio	RN-04, RN-05.
Incluye	Ninguna.
Extiende	Ninguna.

2.3.4.11. CU-11. Hacer pedido

ID	CU-11
Nombre	Hacer pedido
Responsable	Alejandro Chacón
Actualización	23/03/2023
Descripción	Permite al cliente realizar un pedido.
Actor	Cliente (Invitado, miembro).
Disparador	El Cliente da clic en “Incluir”.
Precondiciones	PRE-1. Existe por lo menos un ALIMENTO registrado en el sistema. PRE-2. El Cliente selecciono por lo menos un ALIMENTO del menú.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Agregado”. 2. El sistema actualiza las existencias totales del ALIMENTO. 3. El sistema guarda de manera local los alimentos seleccionados en la sección “Pedido”. 4. El cliente da clic en el “Pedido”. 5. El sistema muestra una tabla con las columnas “Nombre”, “Precio”, “Cantidad”, “Subtotal” de los ALIMENTOS seleccionados, además muestra el campo “Total” con el precio total del PEDIDO y los botones “Confirmar” y “Cancelar”. (Ver FA5.1) 6. El Cliente da clic en el “Confirmar”. 7. El sistema valida que el Cliente que realiza el pedido sea un MIEMBRO. (Ver FA7.1) 8. El sistema recupera de la base de datos la información del MIEMBRO. 9. El sistema registra en el base de datos el PEDIDO asociándolo a la información del cliente y con el estado “En proceso”. (Ver FA9.1, EX01). 10. El sistema muestra el mensaje “Operación exitosa”. 11. Termina el caso de uso.
Flujos alternos	<p>FA5.1. El Cliente sale de la “GUI Menú”.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Si sales del menú se cancelará tu pedido. ¿Estás seguro de salir?” y los botones “Sí” y “No”. 2. El Cliente da clic en “Sí”. 3. El sistema actualiza las existencias totales de los ALIMENTOS que habían sido seleccionados aumentando el número liberado. 4. Termina el caso de uso. <p>FA7.1. El Cliente da clic en el botón “Cancelar”.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “¿Está seguro de que desea cancelar?” y los botones “Sí” y “No”. 2. El Cliente da clic en “Sí”. 3. Termina el caso de uso. <p>FA9.1. El Cliente que realiza el PEDIDO es un invitado.</p>

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

	<ol style="list-style-type: none"> 1. El sistema deshabilita el botón “Confirmar”, debajo de la tabla muestra los campos de texto para que el invitado ingrese su “Dirección”, “Número de teléfono”, “e-mail” y el botón “Guardar”. 2. El invitado ingresa su dirección, número de teléfono y e-mail y da clic en “Guardar”. 3. Va al paso 9 del flujo normal.
Excepciones	<p>EX01. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde” y el botón “Aceptar.” 2. El Cliente da clic en el botón “Aceptar” 3. Termina el caso de uso.
Postcondiciones	<p>POST-1. El PEDIDO se registró de manera exitosa en el sistema.</p> <p>POST-2. Las existencias de los alimentos se actualizaron en tiempo real de forma exitosa.</p>
Reglas de negocio	RN-04, RN-05.
Incluye	Ninguna.
Extiende	Ninguna.

2.3.4.12. CU-12. Ver Historial de Pedidos

ID	CU-12
Nombre	Ver Historial de Pedidos
Responsable	Javier Domínguez
Actualización	23/03/2023
Descripción	Permite al miembro ver el historial de los pedidos que ha realizado.
Actor	Miembro.
Disparador	El miembro da clic en “Historial de pedidos”.
Precondiciones	PRE-1. Existe por lo menos un PEDIDO realizado del MIEMBRO.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema recupera de la base de datos los PEDIDOS realizados por el MIEMBRO. (Ver EX01) 2. El sistema muestra la ventana “GUI Historial de pedidos” con un vista previa de los pedidos realizados mostrando una imagen genérica, la fecha del PEDIDO, su ESTADO y el botón “Detalles”, al final de la ventana muestra el botón “Salir”. 3. El miembro da clic en el botón “Detalles”. 4. El sistema muestra los detalles del PEDIDO seleccionado en una tabla con las columnas “Nombre”, “Descripción”, “Imagen”, “Cantidad”, “Subtotal” de los ALIMENTOS seleccionados, además muestra el campo “Total” con el precio total del PEDIDO y el botón “Volver”. (Ver FA4.1, EX01) 5. El sistema valida que el PEDIDO se encuentre estado “Entregado” o “Cancelado”. 6. El sistema muestra el botón “Pedir de nuevo”. (Ver FA6.1) 7. El miembro da clic en el botón “Pedir de nuevo”. (Ver FA7.1) 8. El sistema valida que estén disponibles las existencias de los ALIMENTOS del PEDIDO. 9. El sistema cambia el ESTADO del pedido a “En proceso”. (Ver FA9.1, EX01) 10. El sistema muestra el mensaje “Operación exitosa”. 11. El miembro da clic en el botón “Volver”. 12. Regresa al Historial de pedidos. 13. El miembro da clic en “Salir”. 14. Termina el caso de uso.
Flujos alternos	<p>FA4.1 El miembro da clic en “Salir”.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “¿Está seguro de que desea cancelar?” y los botones “Sí” y “No”. 2. El miembro da clic en “Sí”. 3. Termina el caso de uso. <p>FA6.1. El pedido se encuentra en un estado diferente a Entregado o Cancelado.</p> <ol style="list-style-type: none"> 1. El miembro da clic en el botón “Volver” 2. Regresa al paso 2 del flujo normal.

	<p>FA7.1. El miembro da clic en “Volver”.</p> <ol style="list-style-type: none"> 1. Va al paso2 del flujo normal. <p>FA9.1. Existencias de los ALIMENTOS del PEDIDO agotadas.</p> <ol style="list-style-type: none"> 1. El sistema muestra en rojo los ALIMENTOS del PEDIDO agotados. 2. El sistema muestra el mensaje “Los alimentos en rojo están agotados. ¿Deseas hacer tu pedido con los alimentos restantes? y los botones “Sí” y “No”. 3. El miembro da clic en “Sí”. 4. El sistema registra un nuevo PEDIDO en la base de datos con estado “En proceso” y los ALIMENTOS disponibles. 5. Va al paso 10 del flujo normal.
Excepciones	<p>EX01. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde” y el botón “Aceptar.” 2. El miembro da clic en el botón “Aceptar” 3. Termina el caso de uso.
Postcondiciones	<p>POST01-El ESTADO del PEDIDO cambio a “En proceso”.</p> <p>POST02-Se registro un nuevo PEDIDO en la base de datos en el caso de que algunos de los ALIMENTOS se encontraron agotados.</p> <p>POST03-El miembro pudo consultar el historial de PEDIDOS de manera exitosa.</p> <p>POST04-Los detalles del PEDIDO se mostraron de forma exitosa.</p>
Reglas de negocio	RN-04, RN-05.
Incluye	Ninguna.
Extiende	Ninguna.

2.3.4.13. CU-13. Gestionar Pedidos

ID	CU-13
Nombre	Gestionar Pedidos
Responsable	Alejandro Chacón
Actualización	30/03/2023
Descripción	Permite al Cajero consultar todos los PEDIDOS registrados en el sistema.
Actor	Cajero.
Disparador	El Cajero da clic en la pestaña “Pedidos”.
Precondiciones	PRE-1. El Cajero autenticó sus credenciales.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema recupera de la base de datos todos los PEDIDOS (ver EX01). 2. El sistema muestra los PEDIDOS en “GUI Pedidos”, mostrando la fecha, hora, monto y estado del pedido. 3. El Cajero da clic en una pestaña distinta (ver FA3.1, FA3.2). 4. Termina el caso de uso.
Flujos alternos	<p>FA3.1. El Cajero da clic en el estado de algún PEDIDO.</p> <ol style="list-style-type: none"> 1. El sistema despliega una lista de posibles estados para esa columna. 2. El cajero selecciona uno de los estados. 3. El sistema asigna el estado seleccionado al PEDIDO correspondiente. 4. Regresa al paso 3 del flujo normal. <p>FA3.2. Un cliente hace un nuevo PEDIDO.</p> <ol style="list-style-type: none"> 1. El sistema actualiza “GUI Pedidos” para mostrar el nuevo PEDIDO. 2. Regresa al paso 3 del flujo normal.
Excepciones	<p>EX01. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde” y el botón “Aceptar.” 2. El Cajero da clic en el botón “Aceptar” 3. Termina el caso de uso.
Postcondiciones	<p>POST-1. El Cajero vió los PEDIDOS.</p> <p>POST-2. El Cajero modificó el estado de un pedido.</p>
Reglas de negocio	RN-04, RN-05.
Incluye	Ninguna.
Extiende	Ninguna.

2.3.4.14. CU-14. Publicar Reseña

ID	CU-14
Nombre	Publicar Reseña
Responsable	Javier Domínguez
Actualización	23/03/2023
Descripción	Permite a un miembro publicar una RESEÑA nueva.
Actor	Miembro.
Disparador	El miembro da clic en “Escribir reseña” en “GUI Reseñas”
Precondiciones	PRE-01. El miembro tiene al menos un pedido pagado.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema muestra “GUI Nueva Reseña” con un campo de texto para los comentarios, un botón para subir imágenes y el botón “Publicar”. 2. El miembro ingresa texto al campo de comentarios y da clic en el botón “Publicar”. 3. El sistema valida que el campo de comentarios esté lleno. 4. El sistema valida que el campo de comentarios tenga sólo valores alfanuméricos (ver FA4.1). 5. El sistema valida que la imagen subida pese un (1) MB o menos, si es que existe alguna (ver FA5.1). 6. El sistema registra la RESEÑA en la base de datos y actualiza la lista de reseñas en todas las instancias de “GUI Reseñas” (ver FA6.1, EX1, EX2). 7. Termina el caso de uso.
Flujos alternos	<p>FA4.1. El miembro dejó el campo de comentarios vacío.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Campos vacíos detectados.” 2. Regresa al paso 2 del flujo normal. <p>FA5.1. El miembro introdujo valores no alfanuméricos en el campo de comentarios.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “Solo se permiten letras y números en: [Nombre del campo]”. 2. Regresa al paso 2 del flujo normal. <p>FA6.1. El miembro intentó subir una imagen con un peso superior a un (1) MB.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “No puede subir imágenes con un peso superior a un (1) MB”. 2. Regresa al paso 2 del flujo normal.
Excepciones	<p>EX1. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No hay conexión con la base de datos. Por favor, inténtalo más tarde” y el botón “Aceptar.” 2. El miembro da clic en el botón “Aceptar” 3. Termina el caso de uso.

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

	<p>EX2. No hay almacenamiento disponible en el servidor.</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje ““No te podemos registrar en este momento. Por favor, inténtalo más tarde” y el botón “Aceptar.” 2. El miembro da clic en el botón “Aceptar” 3. Termina el caso de uso.
Postcondiciones	<p>POST-1. Se ha registrado una nueva RESEÑA en la base de datos.</p> <p>POST-2. Se ha mostrado en tiempo real la nueva RESEÑA a todas las instancias de la aplicación que estén viendo “GUI Reseñas.”</p>
Reglas de negocio	RN-05, RN-06.
Incluye	Ninguna.
Extiende	Ninguna.

2.3.4.15. CU-15. Ver Reseñas

ID	CU-15
Nombre	Ver Reseñas
Responsable	Alejandro Chacón
Actualización	23/03/2023
Descripción	Permite al consultante ver las reseñas publicadas en tiempo real.
Actor	Consultante (cliente/Cajero/Gerente)
Disparador	Consultante da clic en la pestaña de reseñas.
Precondiciones	PRE-1. Hay al menos una RESEÑA registrada en la base de datos.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema recupera todas las RESEÑAS de la base de datos (ver EX1). 2. El sistema valida que el consultante sea Gerente. 3. El sistema despliega las RESEÑAS en “GUI Reseñas”, mostrando la fecha y hora de publicación, los comentarios y la imagen de cada RESEÑA (ver FA3.1). 4. El consultante da clic en otra pestaña (ver FA4.1). 5. Termina el caso de uso (ver FA5.1, FA5.2).
Flujos alternos	<p>FA3.1. El consultante es Gerente</p> <ol style="list-style-type: none"> 1. El sistema muestra el botón “Eliminar” en cada reseña desplegada. 2. Regresa al paso 4 del flujo normal. <p>FA4.1. Un MIEMBRO publica una nueva RESEÑA.</p> <ol style="list-style-type: none"> 1. El sistema actualiza “GUI Reseñas” del consultante y de todas las demás instancias para mostrar la nueva RESEÑA. 2. Regresa al paso 4 del flujo normal. <p>FA5.1. El Gerente da clic en el botón “Eliminar” de alguna reseña.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “¿Está seguro de que desea eliminar la reseña?” con los botones “Sí” y “No”. 2. El Gerente da clic en “Sí” 3. El sistema elimina la RESEÑA de la base de datos (ver EX1). 4. El sistema actualiza todas las instancias de “GUI Reseñas” para remover la RESEÑA eliminada. 5. Regresa al paso 4 del flujo normal. <p>FA5.2. El consultante da clic en “Descargar imagen”</p> <ol style="list-style-type: none"> 1. El sistema guarda la imagen en el dispositivo del consultante (ver EX2 falta de espacio en disco) 2. El sistema muestra el mensaje “Imagen guardada” 3. Regresa al paso 4 del flujo normal.
Excepciones	<p>EX1. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No hay conexión con la base de datos. Por favor, inténtalo más tarde” y el botón “Aceptar.” 2. El consultante da clic en el botón “Aceptar”

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

	<p>3. Termina el caso de uso.</p> <p>EX2. No hay almacenamiento disponible en el dispositivo del consultante.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “No hay espacio disponible en el disco duro” y el botón “Aceptar.” 2. El miembro da clic en el botón “Aceptar” 3. Termina el caso de uso.
Postcondiciones	POST-1. El consultante vió las RESEÑAS publicadas.
Reglas de negocio	RN-05, RN-07.
Incluye	Ninguna.
Extiende	Ninguna.

2.3.4.16. CU-16. Generar Reporte de Pedidos del Día.

ID	CU-16
Nombre	Generar Reporte de Pedidos del Día
Responsable	Javier Domínguez
Actualización	28/03/2023
Descripción	Permite al Gerente ver el reporte de Pedidos del Día.
Actor	Gerente.
Disparador	Gerente da clic en pestaña de Reportes.
Precondiciones	PRE-1. Las credenciales del GERENTE han sido autenticadas.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema despliega “GUI Reporte”, con un seleccionador de fechas. 2. El Gerente selecciona una fecha. 3. El sistema valida que la fecha sea igual o menor a la fecha actual. 4. El sistema recupera todos los PEDIDOS de la base de datos correspondientes a la fecha seleccionada (ver FA4.1, EX1). 5. El sistema actualiza “GUI Reporte”, mostrando una tabla con el ingreso de cada PEDIDO, el ingreso total y la cantidad que se consumió por tipo de ALIMENTO en los PEDIDOS. Incluye el botón “Descargar”. 6. El Gerente cierra la ventana. 7. Termina el caso de uso (ver FA6.1).
Flujos alternos	<p>FA4.1. El gerente seleccionó una fecha futura.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “La fecha debe ser igual o menor a hoy.” 2. Regresa al paso 2 del flujo normal. <p>FA6.1. El Gerente da clic en “Descargar .PDF”</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana de dialogo para elegir la ruta en que se descargará el archivo. 2. El Gerente selecciona la ruta y da clic en “OK”. 3. El sistema descarga el archivo en el dispositivo del Gerente, en la ruta seleccionada (ver EX2). 4. Regresa al paso 5 del flujo normal.
Excepciones	<p>EX1. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No hay conexión con la base de datos. Por favor, inténtalo más tarde” y el botón “Aceptar.” 2. El Gerente da clic en el botón “Aceptar” 3. Termina el caso de uso. <p>EX2. No hay almacenamiento disponible en el dispositivo del Gerente.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “No hay espacio disponible en el disco duro” y el botón “Aceptar.” 2. El miembro da clic en el botón “Aceptar” 3. Termina el caso de uso.
Postcondiciones	POST-1. El consultante vio las RESEÑAS publicadas.

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

Reglas de negocio	Ninguna.
Incluye	Ninguna.
Extiende	Ninguna.

2.3.4.17. CU-17. Generar Reporte de Pedidos del Mes.

ID	CU-17
Nombre	Generar Reporte de Pedidos del Mes.
Responsable	Alejandro Chacón
Actualización	28/03/2023
Descripción	Permite al Gerente ver el reporte de Pedidos del Día.
Actor	Gerente.
Disparador	Gerente da clic en pestaña de Reportes.
Precondiciones	PRE-1. Las credenciales del GERENTE han sido autenticadas.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema despliega “GUI Reporte”, con un seleccionador de fechas. 2. El Gerente selecciona un mes. 3. El sistema valida que el mes seleccionado sea menor al mes actual. 4. El sistema recupera todos los PEDIDOS de la base de datos correspondientes a la fecha seleccionada (ver FA4.1, EX1). 5. El sistema actualiza “GUI Reporte”, mostrando una tabla con el ingreso de cada PEDIDO, el ingreso total y la cantidad que se consumió por tipo de ALIMENTO en los PEDIDOS. Incluye el botón “Descargar”. 6. El Gerente cierra la ventana. 7. Termina el caso de uso (ver FA6.1).
Flujos alternos	<p>FA4.1. El gerente seleccionó una fecha futura.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “La fecha debe ser igual o menor a hoy.” 2. Regresa al paso 2 del flujo normal. <p>FA6.1. El Gerente da clic en “Descargar .PDF”</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana de dialogo para elegir la ruta en que se descargará el archivo. 2. El Gerente selecciona la ruta y da clic en “OK”. 3. El sistema descarga el archivo en el dispositivo del Gerente, en la ruta seleccionada (ver EX2). 4. Regresa al paso 5 del flujo normal.
Excepciones	<p>EX1. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Pérdida de conexión” con el mensaje “No hay conexión con la base de datos. Por favor, inténtalo más tarde” y el botón “Aceptar.” 2. El Gerente da clic en el botón “Aceptar” 3. Termina el caso de uso. <p>EX2. No hay almacenamiento disponible en el dispositivo del Gerente.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “No hay espacio disponible en el disco duro” y el botón “Aceptar.” 2. El miembro da clic en el botón “Aceptar” 3. Termina el caso de uso.
Postcondiciones	POST-1. El consultante vio las RESEÑAS publicadas.

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

Reglas de negocio	Ninguna.
Incluye	Ninguna.
Extiende	Ninguna.

2.3.4.18. CU-18. Dar empleado de alta.

ID	CU-18
Nombre	Dar empleado de alta
Responsable	Javier Domínguez
Fecha de Actualización	28/03/2023
Descripción	Permite al GERENTE registrar a un nuevo EMPLEADO en el sistema.
Actor(es)	Gerente.
Disparador	El Gerente da clic en el botón “Nuevo empleado”.
Precondición(es)	PRE01- Existe por lo menos un tipo de TURNO registrado en el sistema. PRE02- Existe por lo menos un PUESTO registrado en el sistema.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema recupera de la base de datos los TURNOS y PUESTOS. (Ver EX01) 2. El sistema muestra la ventana “GUI Nuevo empleado staff” con una lista que contiene los PUESTOS, una lista con los TURNOS, los campos de texto “Nombre”, “Apellido paterno”, “Apellido materno”, “e-mail”, “Teléfono” y “Dirección” para ingresar los datos del nuevo EMPLEADO y los botones “Registrar” y “Cancelar”. 3. El sistema establece el límite de caracteres en la GUI a un máximo 50 en los campos “Nombre”, “Apellido paterno” y “Apellido materno”. 4. El sistema establece el límite de caracteres en la GUI del “e-mail” en un máximo de 100. 5. El sistema establece el límite de caracteres en la GUI del “Teléfono” en un máximo de 15. 6. El sistema establece el límite de caracteres en la GUI de la “Dirección” en un máximo de 150. 7. El Gerente ingresa los datos del EMPLEADO, selecciona su TURNO, PUESTO y da clic en el botón “Registrar”. (Ver FA7.1) 8. El sistema valida que todos los campos estén llenos o seleccionados. 9. El sistema valida que los campos “Nombre”, “Apellido paterno”, y “Apellido materno” solo contienen letras. (Ver FA9.1) 10. El sistema valida que el campo “Dirección” es de tipo alfanumérico (Ver FA10.1) 11. El sistema valida que el campo “e-mail” tiene el formato de e-mail. (Ver FA11.1) 12. El sistema valida que el campo “Teléfono” es de tipo numérico. (Ver FA12.1) 13. 14. El sistema valida que el EMPLEADO no se encuentre registrado en la base de datos. (Ver FA13.1)

	<p>14. El sistema guarda en la base de datos la información del nuevo EMPLEADO. (Ver FA14.1, EX01)</p> <p>15. El sistema muestra el mensaje “Operación exitosa”.</p> <p>16. El sistema borra la información de los campos de texto que ingreso el Gerente.</p> <p>17. Termina el caso de uso.</p>
Flujos Alternos	<p>FA7.1. El Gerente da clic en el botón “Cancelar”</p> <ol style="list-style-type: none"> 1. El sistema muestra “¿Está seguro de que desea cancelar la operación?” y los botones “Sí” y “No”. 2. El Gerente da clic en el botón “Sí”. 3. Termina el caso de uso. <p>FA9.1. El Gerente no ingreso o no selecciono algún dato.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Todos los campos son obligatorios”. 2. Regresa al paso 7 del flujo normal. <p>FA10.1. El Gerente introdujo valores no alfabéticos en los campos “Nombre”, “Apellido paterno”, y “Apellido materno”.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “Solo se permiten letras en: [Nombre de los campos]”. 2. Regresa al paso 7 del flujo normal. <p>FA11.1. El Gerente introdujo valores no alfanuméricos en el campo “Dirección”.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “Solo se permiten letras y números en: [Nombre del campo]”. 2. Regresa al paso 7 del flujo normal. <p>FA12.1. El campo “e-mail” no tienen formato de e-mail.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “e-mail inválido.” 2. Regresa al paso 7 del flujo normal. <p>FA13.1. El Gerente introdujo valores no numéricos en el campo “Teléfono”.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “Solo se permiten números en: [Nombre del campo]”. 2. Regresa al paso 7 del flujo normal. <p>FA14.1 El EMPLEADO ya se encuentra registrado en la base de datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Empleado existente” con el mensaje “El empleado que desea registrar ya se encuentra en el sistema.” y el botón “Aceptar”. 2. El Gerente da clic en el botón “Aceptar”. 3. Regresa al paso 7 del flujo normal.
Excepciones	<p>EX01.- No hay conexión con la base de datos.</p>

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

	<ol style="list-style-type: none">1. El sistema muestra una ventana emergente “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde...”, y el botón “Aceptar”.2. El Gerente da clic en el botón de “Aceptar”.3. Termina el caso de uso y regresa al menú principal.
Postcondiciones	POST01- La información del nuevo EMPLEADO quedo registrada en el sistema de manera exitosa.
Regla(s) de negocio	RN-08.
Incluye	Ninguno.
Extiende	Ninguno.

2.3.4.19. CU-19. Desplegar lista de empleados.

ID	CU-19
Nombre	Desplegar lista de empleados.
Responsable	Alejandro Chacón
Fecha de Actualización	28/03/2023
Descripción	Permite al GERENTE consultar la lista de EMPLEADOS registrados en el sistema.
Actor(es)	Gerente.
Disparador	El Gerente da clic en el botón “Lista de empleados”.
Precondición(es)	PRE01- Por lo menos hay un EMPLEADO registrado en el sistema con estado “Activo”.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema recupera de la base de datos los EMPLEADOS con estado “Activo” registrados en el sistema. (Ver EX01) 2. El sistema muestra la ventana “GUI Lista de empleados” en una tabla con los encabezados “Nombre completo” y “Puesto”, junto con los botones “Modificar” y “Regresar”. 3. El Gerente da clic en “Regresar”. (Ver FA3.1) 4. Termina el caso de uso.
Flujos Alternos	FA3.1. El Gerente da clic en “Modificar”. <ol style="list-style-type: none"> 1. Extiende al CU20- Modificar empleado 2. El sistema actualiza los registros de la tabla. 3. Regresa al paso 3 del flujo normal
Excepciones	EX01.- No hay conexión con la base de datos. <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde...”, y el botón “Aceptar”. 2. El Gerente da clic en el botón de “Aceptar”. 3. Termina el caso de uso y regresa al menú principal.
Postcondiciones	POST01- El Gerente pudo consultar la lista de EMPLEADOS de manera exitosa.
Regla(s) de negocio	RN-08.
Incluye	Ninguno.
Extiende	CU20- Modificar empleado.

2.3.4.20. CU-20. Modificar empleado

ID	CU-20
Nombre	Modificar empleado.
Responsable	Javier Domínguez
Fecha de Actualización	28/03/2023
Descripción	Permite al GERENTE modificar un EMPLEADO existente en el sistema.
Actor(es)	Gerente.
Disparador	El Gerente da clic en el botón “Modificar”.
Precondición(es)	PRE01- Existe por lo menos un tipo de TURNO registrado en el sistema. PRE02- Existe por lo menos un PUESTO registrado en el sistema. PRE03- El Gerente selecciono un EMPLEADO de la tabla de empelados.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema recupera de la base de datos los TURNOS y PUESTOS. (Ver EX01) 2. El sistema recupera de la base de datos la información de EMPLEADO seleccionado. 3. El sistema muestra la ventana “GUI Nuevo empleado” con una lista que contiene los PUESTOS, una lista con los TURNOS, los campos de texto “Nombre”, “Apellido paterno”, “Apellido materno”, “e-mail”, “Teléfono” y “Dirección” con los datos del EMPLEADO habilitados para cambio y los botones “Guardar”, “Dar de baja” y “Cancelar”. 4. El sistema establece el límite de caracteres en la GUI a un máximo 50 en los campos “Nombre”, “Apellido paterno” y “Apellido materno”. 5. El sistema establece el límite de caracteres en la GUI del “e-mail” en un máximo de 100. 6. El sistema establece el límite de caracteres en la GUI del “Teléfono” en un máximo de 15. 7. El sistema establece el límite de caracteres en la GUI de la “Dirección” en un máximo de 150. 8. El Gerente ingresa nuevos datos del EMPLEADO, selecciona su TURNO, PUESTO y da clic en el botón “Guardar”. (Ver FA8.1, FA8.2) 9. El sistema valida que todos los campos estén llenos o seleccionados. 10. El sistema valida que los campos “Nombre”, “Apellido paterno”, y “Apellido materno” solo contienen letras. (Ver FA10.1) 11. El sistema valida que el campo “Dirección” es de tipo alfanumérico (Ver FA11.1)

	<ol style="list-style-type: none"> 12. El sistema valida que el campo “e-mail” tiene el formato de e-mail. (Ver FA12.1) 13. El sistema valida que el campo “Teléfono” es de tipo numérico. (Ver FA13.1) 14. El sistema valida que no haya otro EMPLEADO registrado en la base de datos con los mismos datos. (Ver FA14.1) 15. El sistema guarda en la base de datos la nueva información del EMPLEADO. (Ver FA15.1, EX01) 16. El sistema muestra el mensaje “Operación exitosa”. 17. Termina el caso de uso.
Flujos Alternos	<p>FA8.1. El Gerente da clic en el botón “Cancelar”</p> <ol style="list-style-type: none"> 1. El sistema muestra “¿Está seguro de que desea cancelar la operación?” y los botones “Sí” y “No”. 2. El Gerente da clic en el botón “Sí”. 3. Termina el caso de uso. <p>FA8.2. El Gerente da clic en el botón “Dar de baja”.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Estas seguro de que deseas dar de baja este empleado” y los botones “Sí” y “No”. 2. El Gerente da clic en “Sí”. 3. Termina el caso de uso. <p>FA10.1. El Gerente no ingreso o no selecciono algún dato.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “Todos los campos son obligatorios”. 2. Regresa al paso 8 del flujo normal. <p>FA11.1. El Gerente introdujo valores no alfabéticos en los campos “Nombre”, “Apellido paterno”, y “Apellido materno”.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “Solo se permiten letras en: [Nombre de los campos]”. 2. Regresa al paso 8 del flujo normal. <p>FA12.1. El Gerente introdujo valores no alfanuméricos en el campo “Dirección”.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “Solo se permiten letras y números en: [Nombre del campo]”. 2. Regresa al paso 7 del flujo normal. <p>FA13.1. El campo “e-mail” no tienen formato de e-mail.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “e-mail inválido.” 2. Regresa al paso 8 del flujo normal.

	<p>FA14.1. El Gerente introdujo valores no numéricos en el campo “Teléfono”.</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje: “Solo se permiten números en: [Nombre del campo]”. 2. Regresa al paso 8 del flujo normal. <p>FA15.1 Existe otro EMPLEADO con los mismos datos registrado en la base de datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI Empleado existente” con el mensaje “Los datos que desea guardar le pertenecen a otro empleado.” y el botón “Aceptar”. 2. El Gerente da clic en el botón “Aceptar”. 3. Regresa al paso 8 del flujo normal.
Excepciones	<p>EX01.- No hay conexión con la base de datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente “GUI Pérdida de conexión” con el mensaje “No se pudo conectar con la base de datos. Por favor, inténtelo más tarde...”, y el botón “Aceptar”. 2. El Gerente da clic en el botón de “Aceptar”. 3. Termina el caso de uso y regresa al menú principal.
Postcondiciones	POST01- La información del nuevo EMPLEADO quedo registrada en el sistema de manera exitosa.
Regla(s) de negocio	RN-08
Incluye	Ninguno.
Extiende	Ninguno.

2.4. Requisitos Funcionales

2.4.1. Requisitos Funcionales Por Caso de Uso

2.4.1.1. CU-01. Ver Menú

- RF-01. El invitado/miembro/Cajero/Gerente puede consultar los alimentos disponibles en tiempo real.
- RF-02. El sistema debe mostrar la información de los alimentos en el menú: el nombre del alimento, la categoría del alimento, el total de existencia, la imagen del alimento, la porción, el precio y una descripción general
- RF-03. El sistema debe mostrar las existencias de cada alimento en tiempo real.
- RF-04. Si un alimento se agota, el sistema debe mostrar el producto como “agotado” e impedir que se pida.
- RF-05. El sistema debe proporcionar la opción para pedir los alimentos del menú.
- RF-06. El sistema debe proporcionar la opción para agregar alimentos al menú.
- RF-07. El sistema debe proporcionar la opción para modificar los alimentos del menú.
- RF-08. El invitado/miembro/Cajero/Gerente puede consultar los detalles de cada alimento.

2.4.1.2. CU-02. Agregar Alimento

- RF-09. El sistema debe proporcionar la opción para agregar un nuevo alimento.
- RF-10. El Gerente puede agregar un nuevo alimento, proporcionando el nombre del alimento, la categoría del alimento, el total de existencia, la imagen del alimento, la porción, el precio y una descripción general.

2.4.1.3. CU-03. Modificar Alimento

- RF-11. El sistema debe proporcionar la opción para modificar un alimento existente.
- RF-12. El Gerente puede modificar un alimento existente, proporcionando el nombre del alimento, la categoría del alimento, el total de existencia, la imagen del alimento, la porción, el precio y/o una descripción general.
- RF-13. El Gerente puede eliminar un alimento existente.

2.4.1.4. CU-04. Agregar Categoría de Alimento

- RF-14. El Gerente puede agregar una nueva categoría de alimento.
- RF-15. El Gerente puede agregar una nueva categoría de alimento proporcionando su nombre, y la forma en que se mide su porción (unidad, mililitros o gramos)

2.4.1.5. CU-05. Actualizar Categoría de Alimento

- RF-16. El Gerente puede actualizar una categoría de alimento existente.
- RF-17. El Gerente puede actualizar una categoría de alimento proporcionando su nombre y/o la forma en que se mide su porción (unidad, mililitros o gramos).

2.4.1.6. CU-06. Publicar Oferta

- RF-18. El Gerente puede publicar una nueva oferta, proporcionando el porcentaje de descuento, la fecha de inicio y fin, y los alimentos en los que aplica.
- RF-19. El sistema debe enviar un e-mail a los miembros consentidos (20 o más pedidos), notificándoles acerca del descuento del que son partícipes y el periodo de tiempo en el que aplica.

2.4.1.7. CU-07. Consultar Lista de Ofertas

RF-20. El Gerente puede desplegar todas las ofertas existentes, viendo su porcentaje de descuento, alimentos participantes, fechas de inicio y de fin.

2.4.1.8. CU-08. Renovar Oferta

RF-21. El Gerente puede publicar de nuevo una oferta existente, modificando la fecha de inicio y fin.

RF-22. El Gerente puede modificar una oferta existente inactiva, proporcionando el porcentaje de descuento, y los alimentos en los que aplica.

RF-23. RF19

2.4.1.9. CU-09. Volverse Miembro

RF-24. El Invitado puede volverse miembro del sistema proporcionando su e-mail, teléfono, nombre completo, dirección y contraseña.

RF-25. Posterior a su registro, un miembro del sistema es candidato a ser miembro consentido, dependiendo de su consumo.

2.4.1.10. CU-10. Actualizar Perfil

RF-26. El miembro puede modificar su nombre completo, e-mail, teléfono, y contraseña.

RF-27. El sistema debe solicitar la contraseña actual del miembro antes de cambiarla por una nueva.

RF-28. El sistema debe enviar al miembro un e-mail con un código de verificación en el caso de que el miembro requiera cambiar su e-mail para iniciar sesión.

2.4.1.11. CU-11. Hacer pedido

RF-29. El sistema debe mostrar los alimentos seleccionados del menú por el miembro, agrupándolos por la cantidad ordenada de cada alimento, el precio de cada alimento según la cantidad ordenada, y el costo total del pedido.

RF-30. El miembro puede agregar alimentos del menú a su pedido.

RF-31. El sistema debe actualizar las existencias de cada alimento en tiempo real, conforme los miembros introduzcan dichos alimentos a sus pedidos correspondientes.

2.4.1.12. CU-12. Ver Historial de Pedidos

RF-32. El miembro puede ver su historial de pedidos.

RF-33. El miembro puede borrar de su historial el registro de los pedidos que haya realizado.

RF-34. El miembro puede volver a hacer un pedido desde su historial de pedidos.

2.4.1.13. CU-13. Gestionar Pedidos

RF-35. El Cajero puede visualizar todos los pedidos registrados en el sistema.

RF-36. Si un nuevo pedido es realizado, el sistema debe actualizar la lista de pedidos en tiempo real.

RF-37. El Cajero puede modificar el estado de un pedido (“En proceso”, “Enviado” y “Entregado”).

RF-38. El Cajero puede cancelar la orden del cliente.

2.4.1.14. CU-14. Publicar reseña

RF-39. El miembro puede publicar una reseña ingresando los comentarios de esta.

RF-40. El miembro puede subir imágenes a su reseña.

2.4.1.15. CU-15. Ver reseñas

- RF-41. El cliente puede consultar las reseñas de los demás clientes.
- RF-42. El sistema debe actualizar las reseñas en tiempo real.
- RF-43. El cliente puede descargar imágenes incluidas en las reseñas.
- RF-44. El gerente puede eliminar reseñas.

2.4.1.16. CU-16. Generar Reporte de Pedidos del Día

- RF-45. El Gerente puede consultar el reporte de ventas a domicilio del día, donde se incluyen todos los pedidos del día, el ingreso monetario por pedido, el ingreso total, y los consumos por tipo de alimento.
- RF-46. El Gerente puede descargar el reporte generado en formato .PDF.

2.4.1.17. CU-17. Generar Reporte de Pedidos del Mes.

- RF-47. El Gerente puede consultar el reporte de ventas a domicilio del mes, donde se incluyen todos los pedidos del mes, el ingreso monetario por pedido, el ingreso total, y los consumos por tipo de alimento.
- RF-48. El Gerente puede descargar el reporte generado en formato PDF.

2.4.1.18. CU-18. Dar empleado de alta

- RF-49. El Gerente puede dar de alta a un empleado, proporcionando su nombre, apellido paterno, apellido materno, teléfono, e-mail, dirección, tipo de turno y puesto.

2.4.1.19. CU-19. Desplegar lista de empleados.

- RF-49. El Gerente puede consultar la lista general de empleados en estado “Activo” que contenga el nombre del empleado y el puesto que ejerce.

2.4.1.20. CU-20. Modificar empleado

- RF-50. El Gerente puede modificar los datos de un empleado, ingresando el nuevo nombre, apellido paterno, apellido materno, teléfono, e-mail, dirección, puesto y tipo de turno.
- RF-51. El Gerente puede dar de baja a un empleado.

2.4.2. Requisitos Funcionales Generales

- RFG-1. El sistema debe mostrar el aviso de campos vacíos en aquellos casos en los que sea necesario llenar un formulario.
- RFG-2. El sistema debe informar cuando haya una pérdida de conexión con la base de datos.
- RFG-3. El sistema debe detener el registro de información textual que contenga caracteres ilegales.
- RFG-4. El sistema debe mostrar un mensaje de confirmación cuando algún dato vaya a ser borrado o modificado por el usuario.
- RFG-5. El sistema debe mostrar el mensaje “Operación exitosa” al completar correctamente un registro o modificación en la base de datos.
- RFG-6. El sistema debe mostrar el mensaje “Campos vacíos detectados” en caso de que se intente registrar cualquier formulario con campos vacíos.
- RFG-7. El sistema debe mostrar el mensaje “Solo se permiten letras y números en: [Nombre del campo]” en caso de que se intente registrar un campo que sea exclusivamente alfanumérico.
- RFG-8. El sistema debe mostrar el mensaje “Solo se permiten números en: [Nombre del campo]” en caso de que se intente registrar un campo que sea exclusivamente numérico.

- RFG-9. El sistema debe mostrar el mensaje “Solo se permiten letras en: [Nombre del campo]” en caso de que se intente registrar un campo que sea exclusivamente alfabético.
- RFG-10. El sistema debe mostrar el mensaje “No puede subir imágenes con un peso superior a un (1) MB” en caso de que se intente subir una imagen con un peso superior a 1 MB.

2.4.3. Requisitos No Funcionales

A continuación, se describen los atributos de calidad que, durante la etapa de Requerimientos, se han contemplado como importantes para el desarrollo de T.A.C.O.S. Lo anterior está sujeto a cambios durante la fase de Diseño Arquitectónico, donde se formalizará tanto la elección de atributos de calidad, como las tácticas para satisfacerlos.

2.4.3.1. Desempeño

- RNF-1. Todas las pantallas de la interfaz gráfica deberán cargarse en 5 segundos o menos.
- RNF-2. Todas las consultas deberán tomar 6 segundos o menos.

2.4.3.2. Disponibilidad

- RNF-3. La API deberá utilizar la táctica *Control de Cordura (Sanity Checking)* para validar las entradas de datos (Bass, Clements y Kazman, 2021).
- RNF-4. El cliente del API deberá utilizar la táctica *Ping* cada 30 segundos para asegurarse que el servidor del API sigue operacional (Bass, Clements y Kazman, 2021).

2.4.3.3. Seguridad

- RNF-5. Las contraseñas deberán cifrarse utilizando el algoritmo AES.

2.4.3.4. Facilidad de Modificación

- RNF-6. La construcción del API deberá priorizar el bajo acoplamiento, prefiriendo tácticas como la *Encapsulación y Abstracción de Servicios Comunes* (Bass, Clements y Kazman, 2021).
- RNF-7. La construcción del sistema deberá seguir un estándar de programación, el cual implemente prácticas de escritura de código que faciliten su lectura y modificación.

2.5. Restricciones

- REST-01 El sistema debe implementarse siguiendo una arquitectura distribuida para garantizar la alta disponibilidad.
- REST-02 Es necesario utilizar diferentes tecnologías para el API y la aplicación cliente.
- REST-03 El sistema debe soportar el manejo de contenido multimedia.
- REST-04 El sistema debe soportar el manejo de información en un ambiente concurrido.

3. Diseño Arquitectónico

3.1. Introducción

En este documento se modela la arquitectura del sistema T.A.C.O.S la cual tiene un enfoque basado en componentes basado en 4 flujos de trabajo: Requerimientos, Especificación, Provisionamiento y Ensamblado, por medio de ella podremos determinar la estructura del sistema para su posterior construcción. Este proceso comienza desde la definición de los procesos del negocio hasta la arquitectura de componentes final y por medio de un refinamiento es posible

llegara a las interfaces que serán las funcionalidades ofrecidas por el sistema a crear junto con la relación entre ellas estableciendo así las dependencias entre componentes y un mapeo de la conexión necesaria para que cada componentes funcione conforme a lo descrito en la firma de los métodos de sus interfaces (el contrato de grano más fino entre dos partes).

3.2. Definición de Requerimientos

El proceso de desarrollo del diseño de la arquitectura del sistema T.A.C.O.S se divide en cuatro flujos de trabajo, comenzando por el flujo de definición de requerimientos, el cual se encarga de generar el modelo de casos de uso y el modelo de conceptos del negocio. Estos artefactos servirán como las bases para el resto del proceso.

3.3. Modelo de Procesos del Negocio

En esta subsección se modelan los distintos diagramas de los procesos del negocio para aquellos procesos no triviales necesarios de documentar, los cuales serán base para los posteriores diagramas e información que el sistema necesita conocer y manejar.

3.3.1. Modelo de proceso “Ordenar In Situ”

El siguiente diagrama muestra el modelo del proceso de negocio para ordenar In Situ. El proceso inicia cuando el cliente llega al establecimiento, posteriormente el mesero le proporciona el menú para que lo inspeccione, cuando el cliente ha inspeccionado el menú procede a realizar su pedido (proceso “Realizar pedido”). Terminado lo anterior si el cliente no paga el pedido el proceso termina, en caso contrario, el mesero entrega el pedido al cliente.

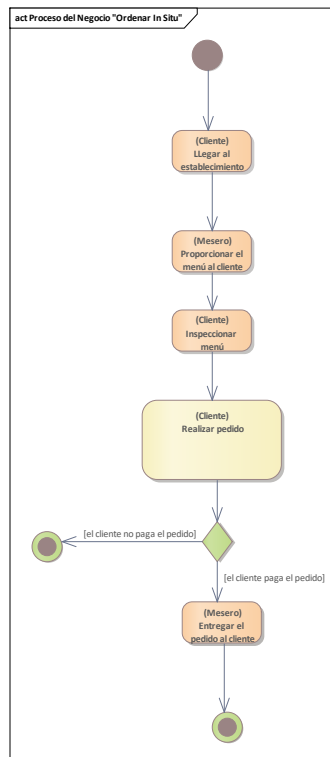


ILUSTRACIÓN 9 PROCESO DEL NEGOCIO “ORDENAR IN SITU”

3.3.2. Modelo de proceso “Ordenar Remoto”

El siguiente diagrama de actividad modela el proceso para ordenar de forma remota un pedido, el cual comienza cuando el cliente llama al establecimiento, el mesero contesta la llamada para que posteriormente el cliente le solicite su pedido (proceso “Realizar pedido”), en caso de que no conozca el menú el cliente pregunta sobre los alimentos disponibles, el mesero le explica cuál es el menú para que posteriormente el cliente realice su pedido (proceso “Realizar pedido”), para poder enviar la orden el mesero solicita los datos personales del cliente y entrega el pedido al repartidor. Si al buscar al cliente el repartidor no lo encuentra éste devuelve el pedido al mesero, en caso contrario cobra el costo total del pedido al cliente, si el cliente paga, el repartidor le entrega el pedido, de no ser así devuelve el pedido al mesero.

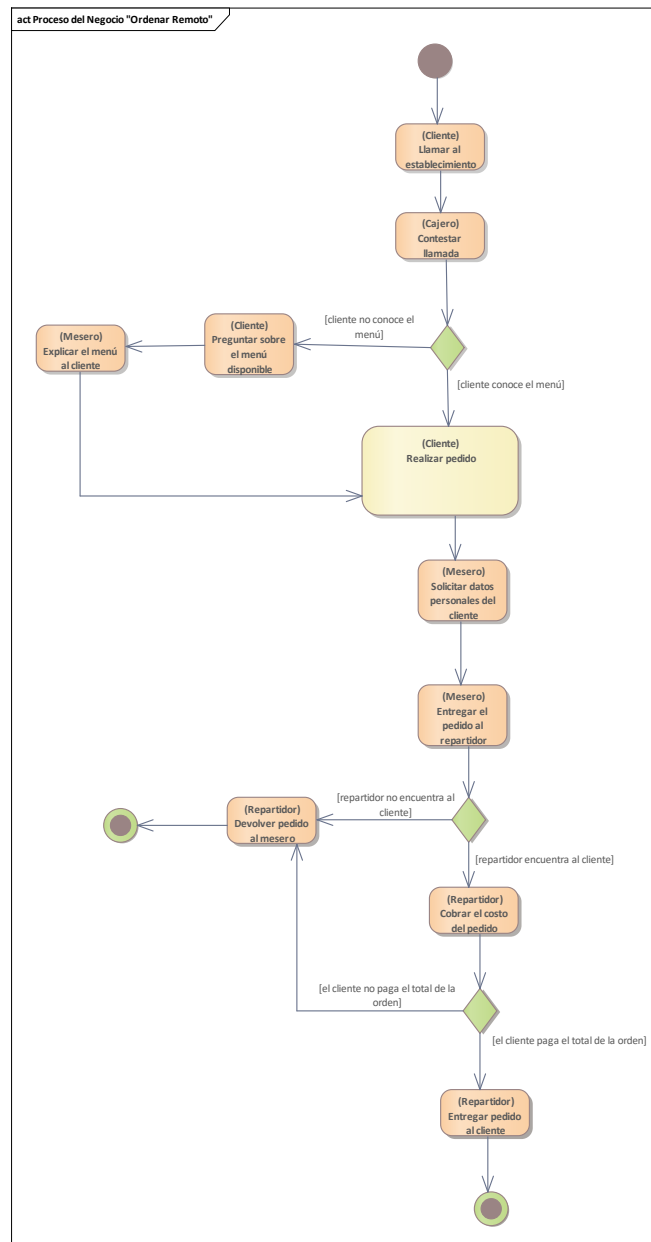


ILUSTRACIÓN 10 PROCESO DEL NEGOCIO “ORDENAR REMOTO”

3.3.3. Modelo de proceso “Realizar pedido”

El siguiente diagrama muestra el modelo del proceso de negocio “Realizar pedido”. El proceso comienza con el cliente seleccionado los alimentos de su pedido para que el mesero los anote. El mesero se dirige con el taquero para consultar si todos los alimentos del pedido están disponibles o ya se agotaron, si todos los alimentos están disponibles el taquero prepara la orden solicitada, en caso de que no sea así el mesero pregunta al cliente si su orden puede no contener los alimentos faltantes. En el escenario de que el cliente acepte la orden incompleta, el taquero comienza a preparar la orden, de no ser de este modo, termina el proceso.

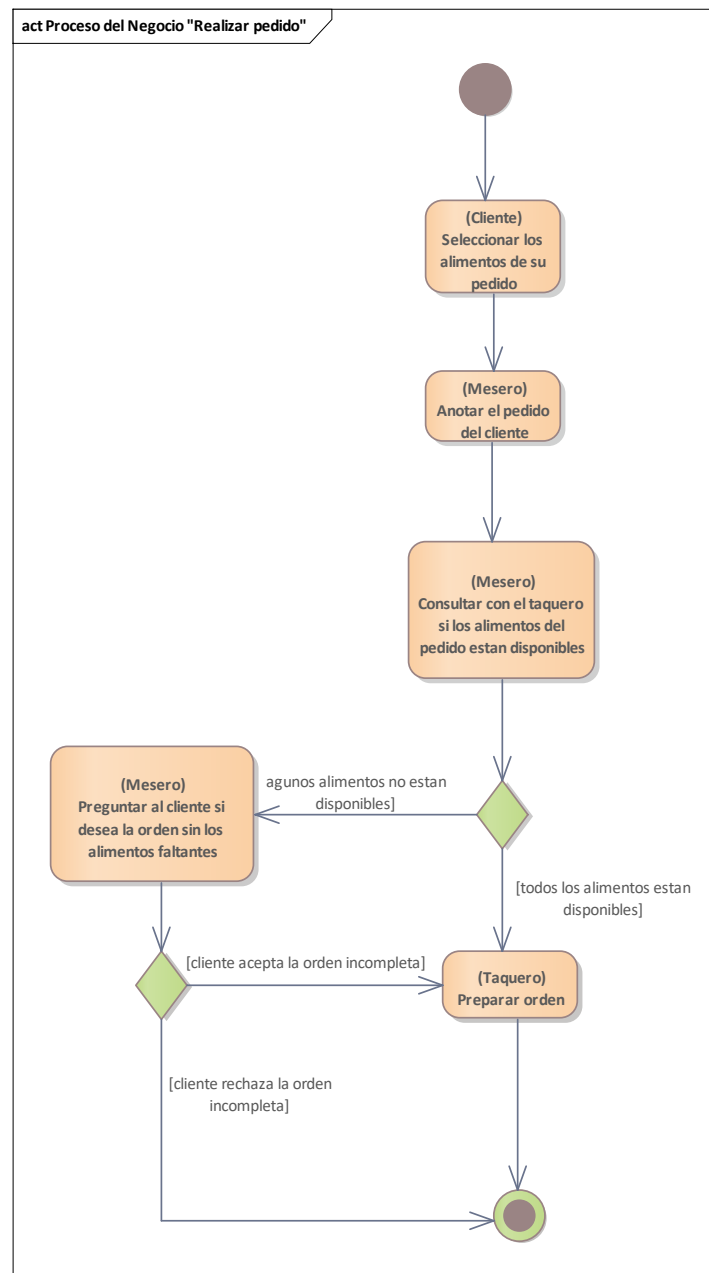


ILUSTRACIÓN 11 PROCESO DEL NEGOCIO “REALIZAR PEDIDO”

3.3.4. Modelo de proceso “Contratar empleado”

El diagrama mostrado a continuación expone el modelo de proceso “Contratar empleado” el cual comienza con la llegada de los candidatos al establecimiento para posteriormente presentar sus solicitudes de empleo. Una vez ocurrido lo anterior el gerente recibe las solicitudes y si el candidato no cumple con las características deseadas procede a rechazar sus solicitud, por el contrario, si cumple con las características acepta la solicitud, llama al candidato y lo entrevista, en caso de que posterior a la entrevista se considera que el candidato no es apto para el puesto es rechazado, de otro modo es contratado y el gerente le asigna el puesto que solicitó.

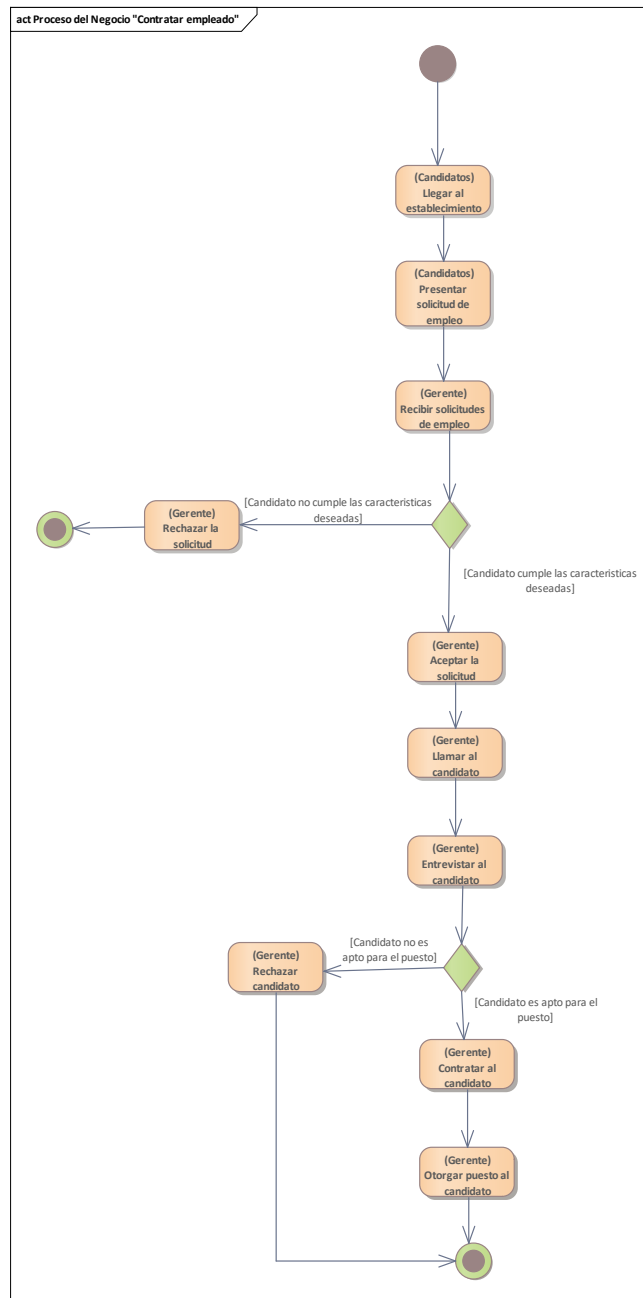


ILUSTRACIÓN 12 PROCESO DEL NEGOCIO “CONTRATAR EMPLEADO”

3.4. Modelo de Conceptos del Negocio

Mediante el modelado de los procesos del negocio, y su posterior análisis, se llegó a un modelo de conceptos del negocio en el cual se plasman los distintos términos del dominio del negocio necesarios para diseñar la arquitectura de TACOS, así como la forma en que se relacionan entre ellos.

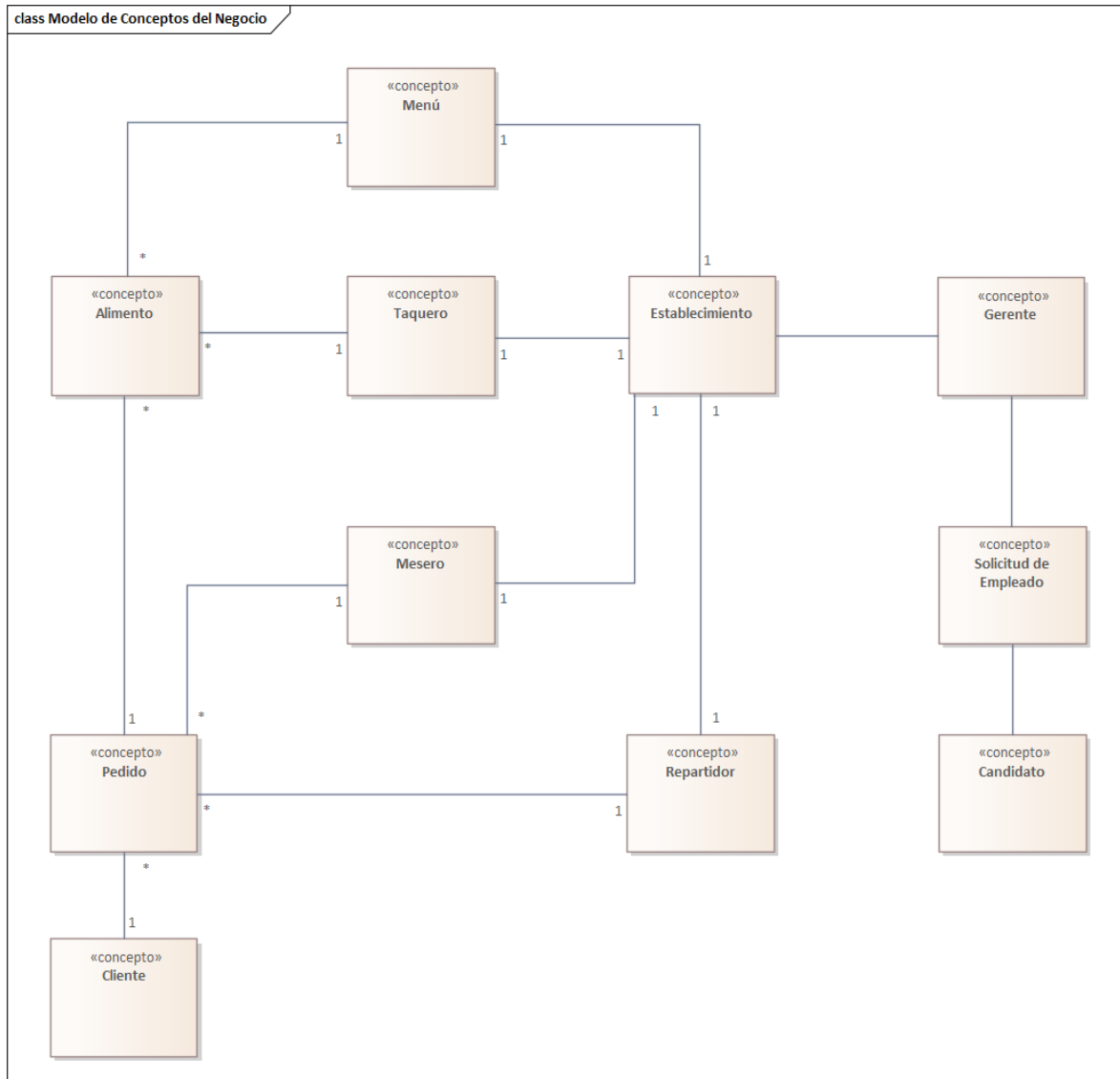


ILUSTRACIÓN 13 MODELO DE CONCEPTOS DEL NEGOCIO

3.5. Visión del Sistema

Se requiere un sistema para la gestión de ventas de comida rápida (tacos) a domicilio. Actualmente, el negocio maneja estas transacciones manualmente; un cliente llama por teléfono al establecimiento, donde un empleado es el encargado de contestar, dictar el menú disponible y tomar el pedido del cliente, mismo que es preparado por un y entregado por un repartidor. Se busca que el producto software propuesto, TACOS, facilite este proceso al sistematizarlo, permitiendo que la clientela del establecimiento pueda consultar y pedir sus platillos favoritos mediante un sistema distribuido, el cual les permita consultar el menú disponible y sus existencias en tiempo real, mantener su historial de pedidos, y publicar reseñas sobre el servicio recibido. Por la parte del *staff* del establecimiento, el sistema les permitirá mantener un registro de los clientes frecuentes (*consentidos*) para ofrecerles ofertas en agradecimiento por su preferencia; gestionar el menú del restaurant y consultar un concentrado de las ventas a domicilio, ya sean diarias o mensuales.

3.6. Tabla de Verificación de Creación y Destrucción

En esta sección, se auditan todos los conceptos del negocio descubiertos en la sección 1.2, con el propósito de establecer cuáles de ellos contienen información que debe ser registrada, modificada y/o removida.

Concepto	Justificación	Creado	Actualizado	Eliminado
Alimento	Deben poderse crear, actualizar y eliminar para gestionar el menú.	✓	✓	✓
Cliente	El registro de sus datos es importante para el proceso de ordenar, por lo que su creación y actualización es necesaria. La eliminación se saca del alcance a petición del negocio.	✓	✓	
Establecimiento	Fuera del alcance.			
Empleado	Un empleado se crea al contratar a un candidato. Los empleados pueden ser modificados, ya sea para cambiar datos personales, o su puesto en la organización. La eliminación se saca del alcance a petición del negocio.	✓	✓	
Menú	El menú es ubicuo, pero su modificación es implícita al crear, actualizar y eliminar alimentos.		✓	
Mesero	Fuera del alcance.			
Pedido	Un cliente realiza pedidos, y el <i>staff</i> encargado de gestionarlos deberá poder modificarlos, para confirmar si ya fueron pagados. La eliminación se saca del alcance a petición del negocio.	✓	✓	
Repartidor	Fuera del alcance.			
Taquero	Fuera del alcance.			

3.7. Tabla de Verificación de Actualización de Relaciones

A continuación, se hace un desglose de las relaciones entre los conceptos del Modelo de Conceptos del Negocio, en el que se especifica si ésta debe cambiar, así como la razón y/o momento en la que esto deba suceder.

Para evitar redundancia, se omiten relaciones que incluyan conceptos que hayan quedado fuera del alcance de TACOS (i.e. Establecimiento, Mesero, Repartidor y Taquero).

Concepto	Justificación
Menú-Alimento	Nunca cambia. Todos los alimentos pertenecen al mismo menú, aunque cambien los datos de algún alimento en específico.
Alimento-Pedido	Nunca cambia.
Menú	Su modificación es implícita al crear, actualizar y eliminar alimentos.
Mesero	Fuera del alcance.
Pedido	Un cliente realiza pedidos, y el <i>staff</i> encargado de gestionarlos deberá poder modificarlos, para confirmar si ya fueron pagados. La eliminación se saca del alcance a petición del negocio.
Repartidor	Fuera del alcance.
Taquero	Fuera del alcance.

3.8. Modelo de Paquetes de Casos de Uso

A continuación, se define el segundo producto del flujo de trabajo de requerimientos; el modelo de casos de uso, comenzando por el empaquetado de los casos de uso. Mediante un análisis de los casos de uso encontrados se definieron cuatro paquetes de casos de uso principales aunado al paquete de actores. A continuación, se presentarán los paquetes de casos: “Manejo de Clientes”, “Gestión de Pedidos”, “Control de Personal” y “Administración del Menú”, además del paquete “Actores”.

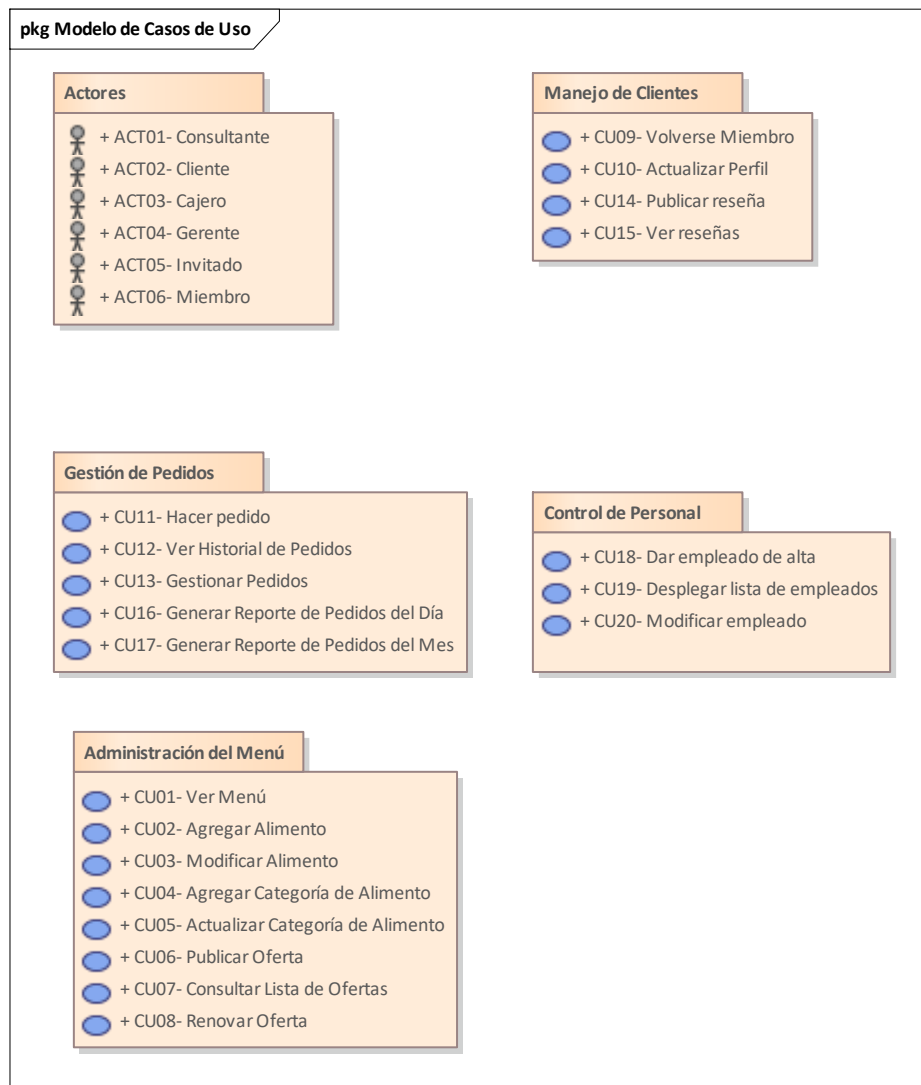


ILUSTRACIÓN 14 MODELO DE PAQUETES DE CASOS DE USO "TACOS"

3.8.1. Modelo de casos de uso: Paquete “Manejo de clientes”

El siguiente diagrama muestra los casos de uso del paquete “Manejo de clientes” para la gestión de los clientes dentro del sistema.

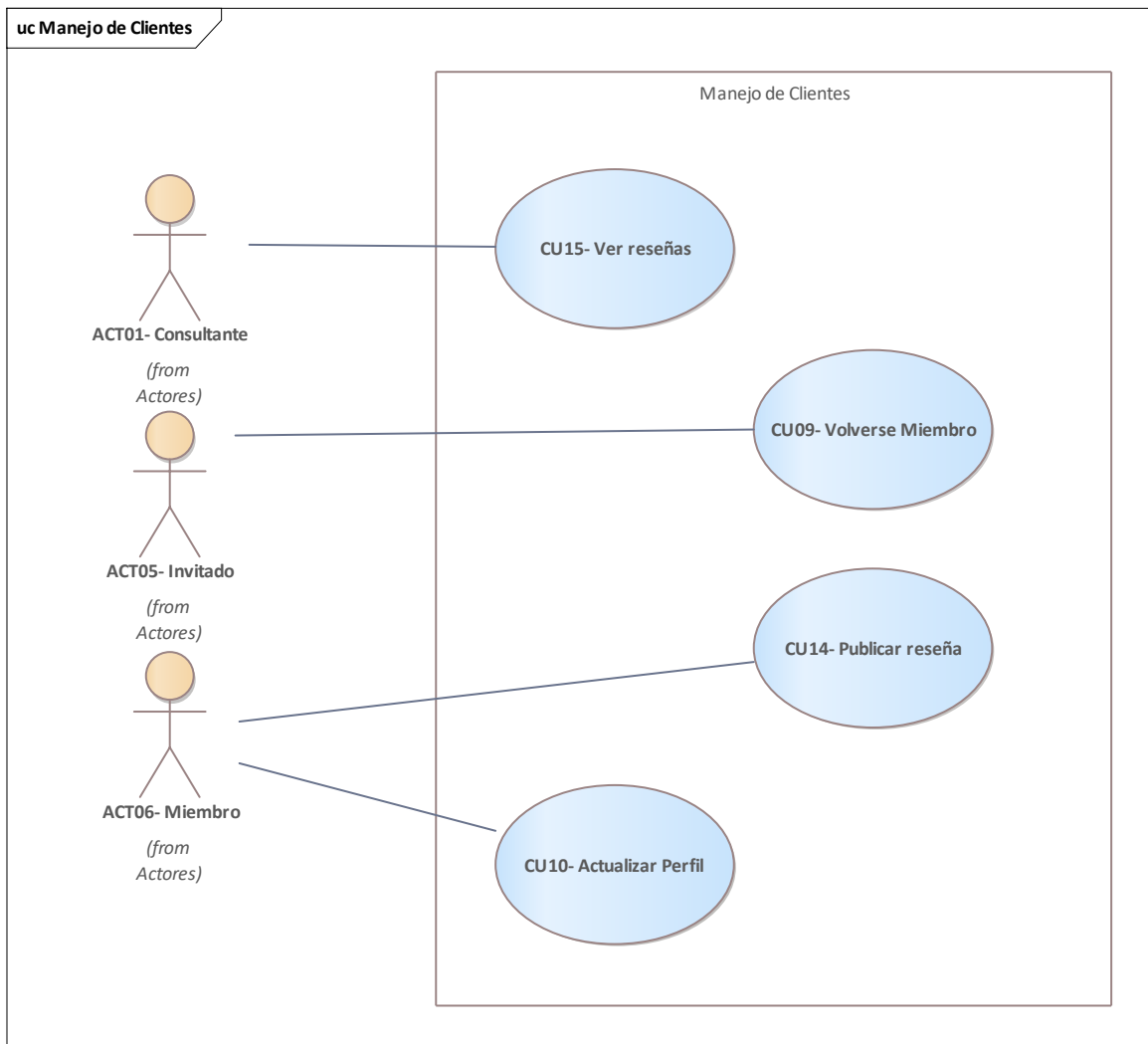


ILUSTRACIÓN 15 MODELO DE CASOS DE USO DEL PAQUETE “MANEJO DE CLIENTES”

3.8.2. Modelo de casos de uso: Paquete “Gestión de pedidos”

El siguiente diagrama muestra los casos de uso del paquete “Gestión de pedidos” para la administración de los aspectos relacionados a los pedidos dentro del sistema.

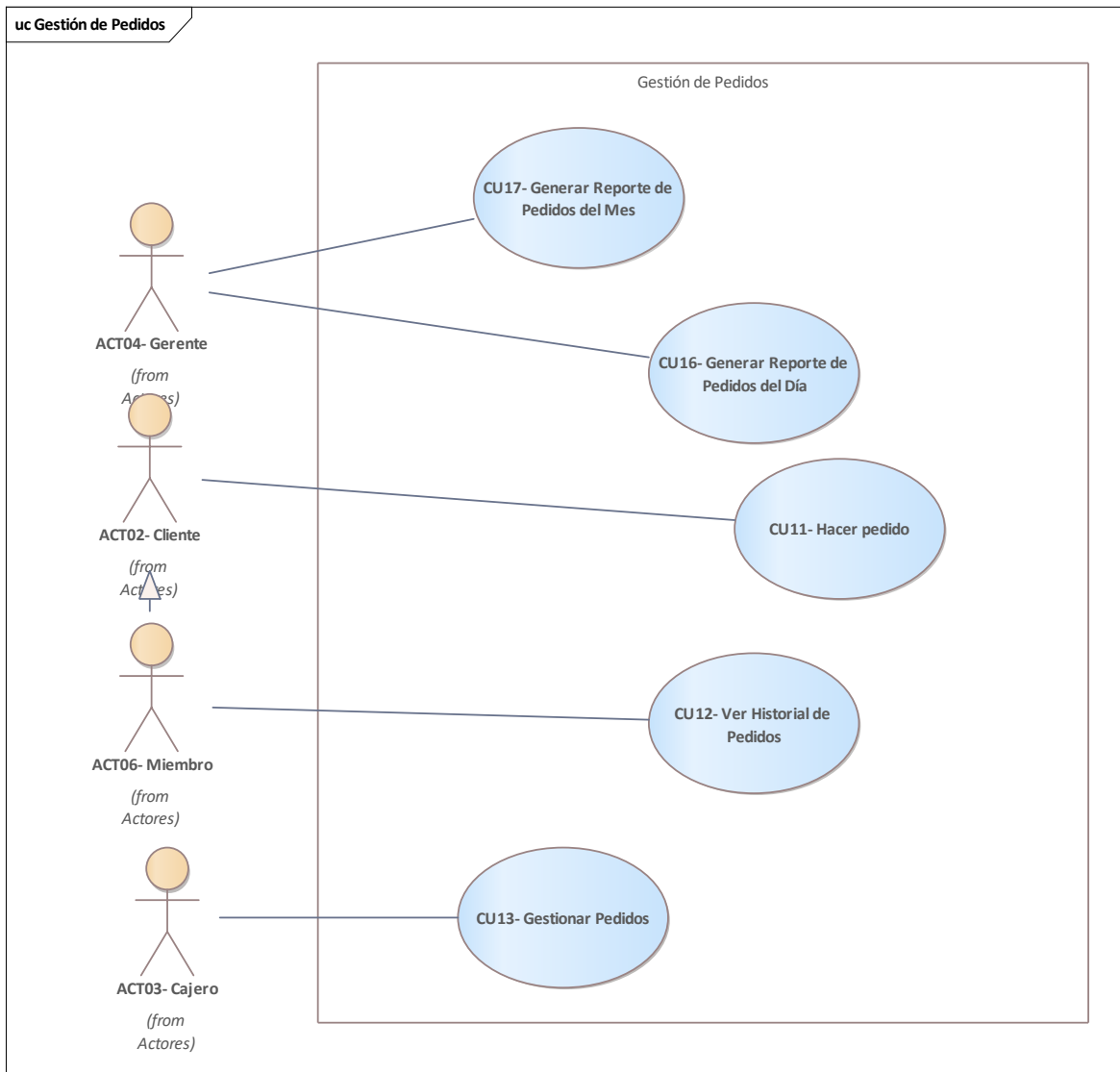


ILUSTRACIÓN 16 MODELO DE CASOS DE USO DEL PAQUETE “GESTIÓN DE PEDIDOS”

3.8.3. Modelo de casos de uso: Paquete “Control de personal”

El siguiente diagrama expone los casos de uso del paquete “Control de personal” para la gestión de los miembros del personal dentro del sistema.

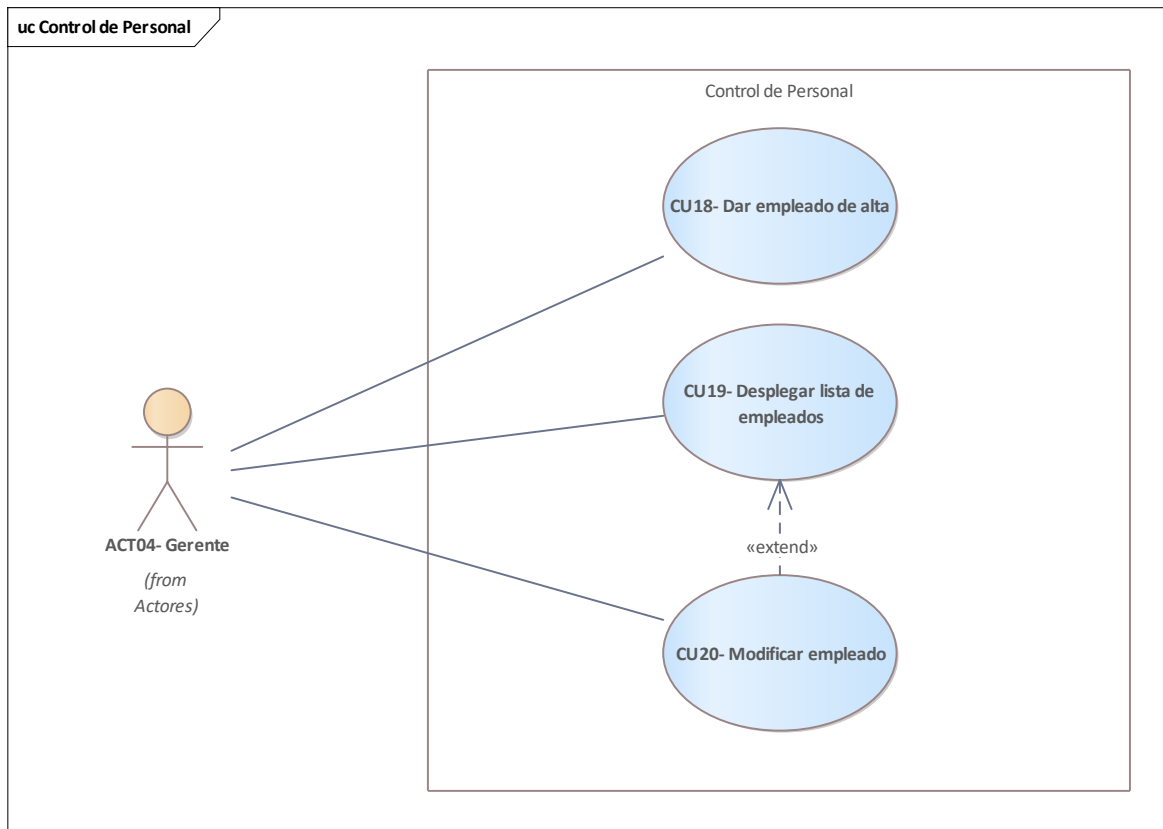


ILUSTRACIÓN 17 MODELO DE CASOS DE USO DEL PAQUETE “MANEJO DE PERSONAL”

3.8.4. Modelo de casos de uso: Paquete “Administración del Menú”

El siguiente diagrama muestra los casos de uso del paquete “Gestión de pedidos” para el control del menú dentro del sistema.

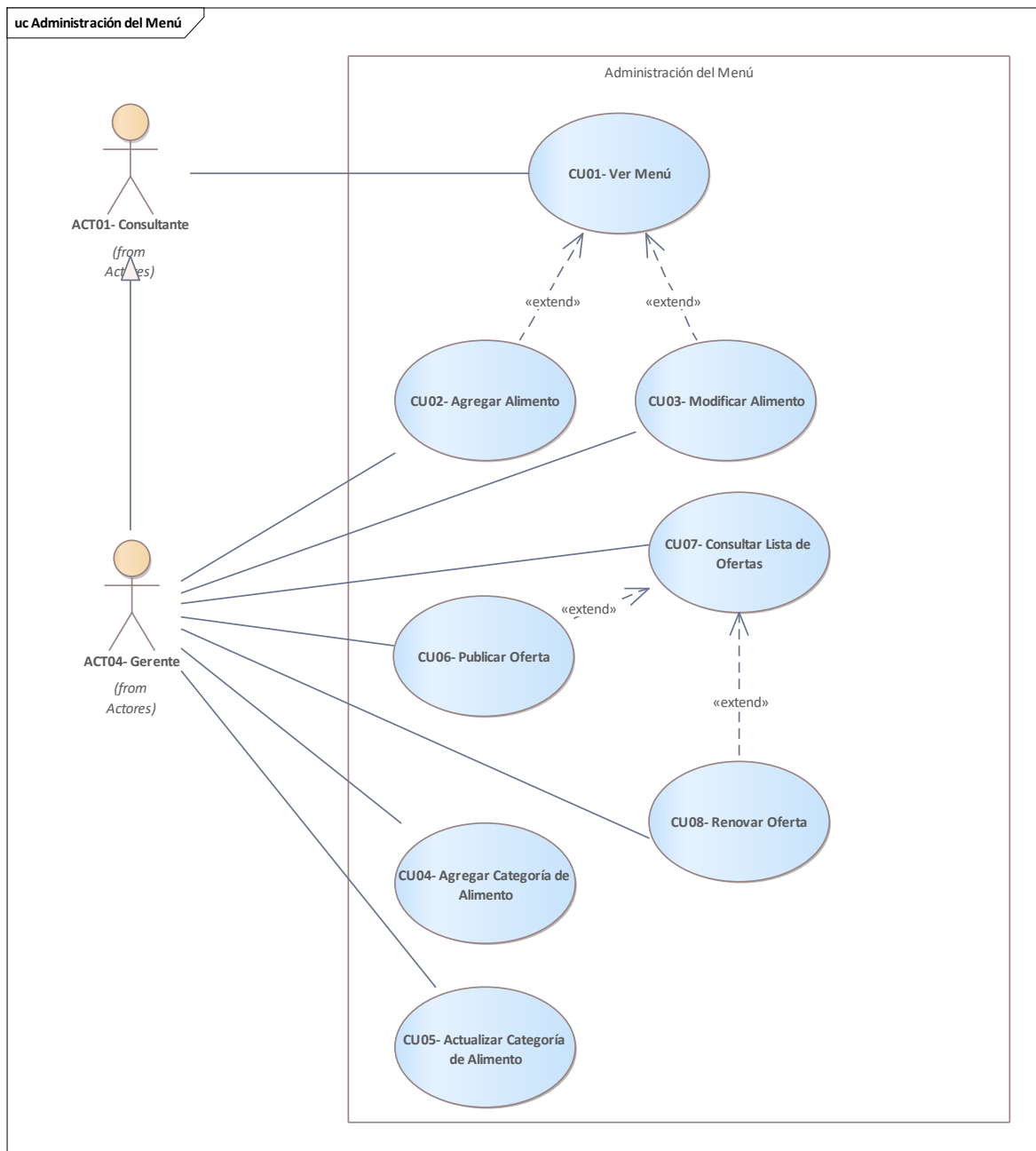


ILUSTRACIÓN 18 MODELO DE CASOS DE USO DEL PAQUETE “ADMINISTRACIÓN DEL MENÚ”

Identificación de Componentes

El flujo de trabajo de identificación se divide en múltiples fases; la refinación del modelo de conceptos del negocio para descubrir los *tipos* del negocio, el descubrimiento de las interfaces del negocio mediante el análisis de tales tipos, el hallazgo de las interfaces del sistema mediante el análisis de los casos de uso y, finalmente, la especificación de los componentes que implementan las interfaces del negocio y sistema descubiertas. Todo se incorpora en la *Arquitectura Inicial de Componentes*.

3.9. Modelo de Tipos de Negocio

El siguiente diagrama es el resultado de refinar el modelo de tipos de negocio para definir con exactitud los conjuntos de información del negocio, o *tipos*, que entran en el alcance del sistema, y la relación que existe entre estos tipos.

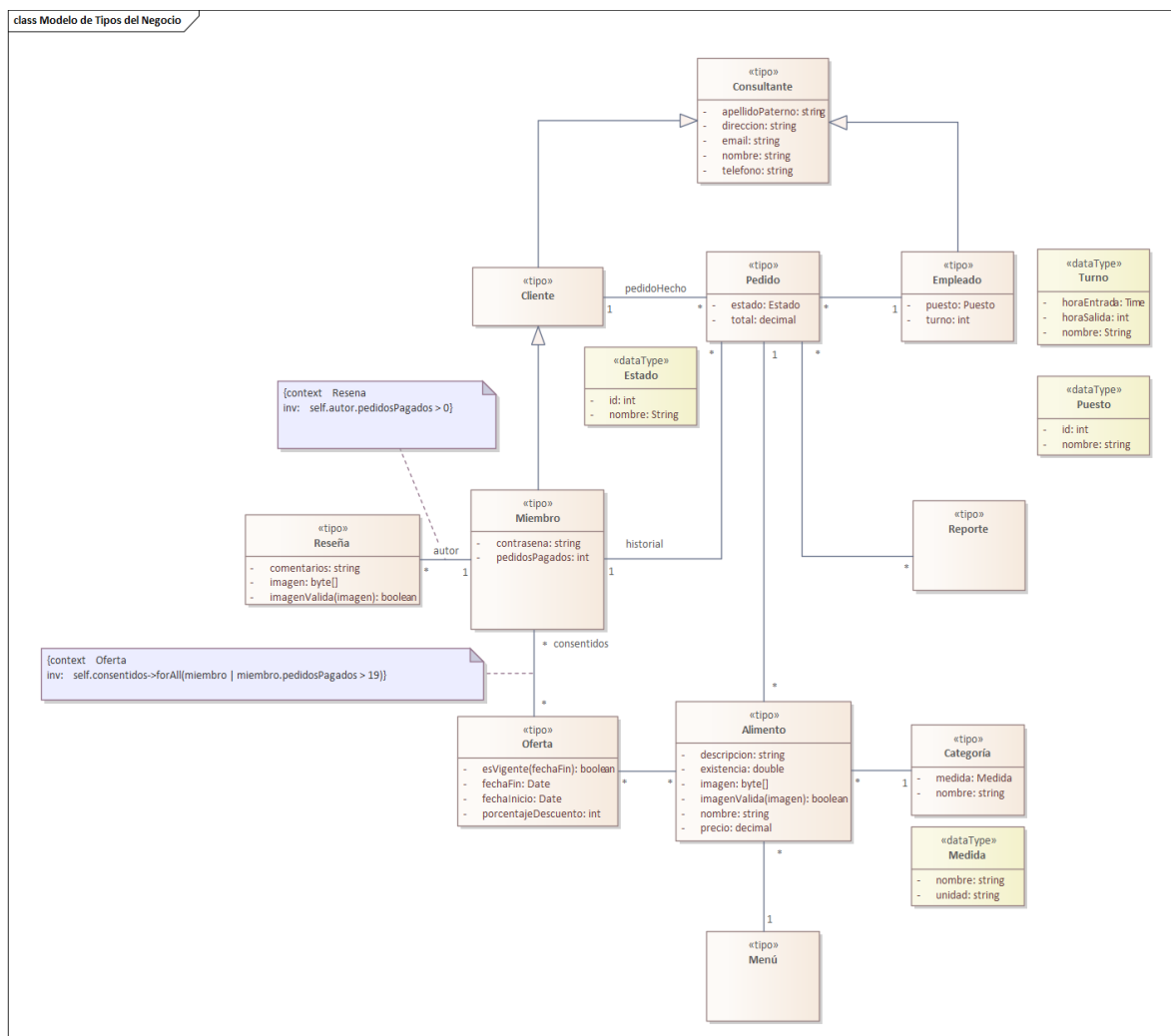


ILUSTRACIÓN 19 MODELO DE TIPOS DEL NEGOCIO

3.10. Modelo de Responsabilidades de Interfaces e Interfaces del Negocio

Se descubren aquellos tipos cuya existencia es independiente del resto de los tipos – *core* –, y se les asigna una interfaz del negocio – *interface type* –, la cual se encargará de gestionar la información de todos los core que le pertenecen, así como todos los tipos que dependen del core asignado.

Lo anterior se plasma en el siguiente diagrama de responsabilidad de interfaz.

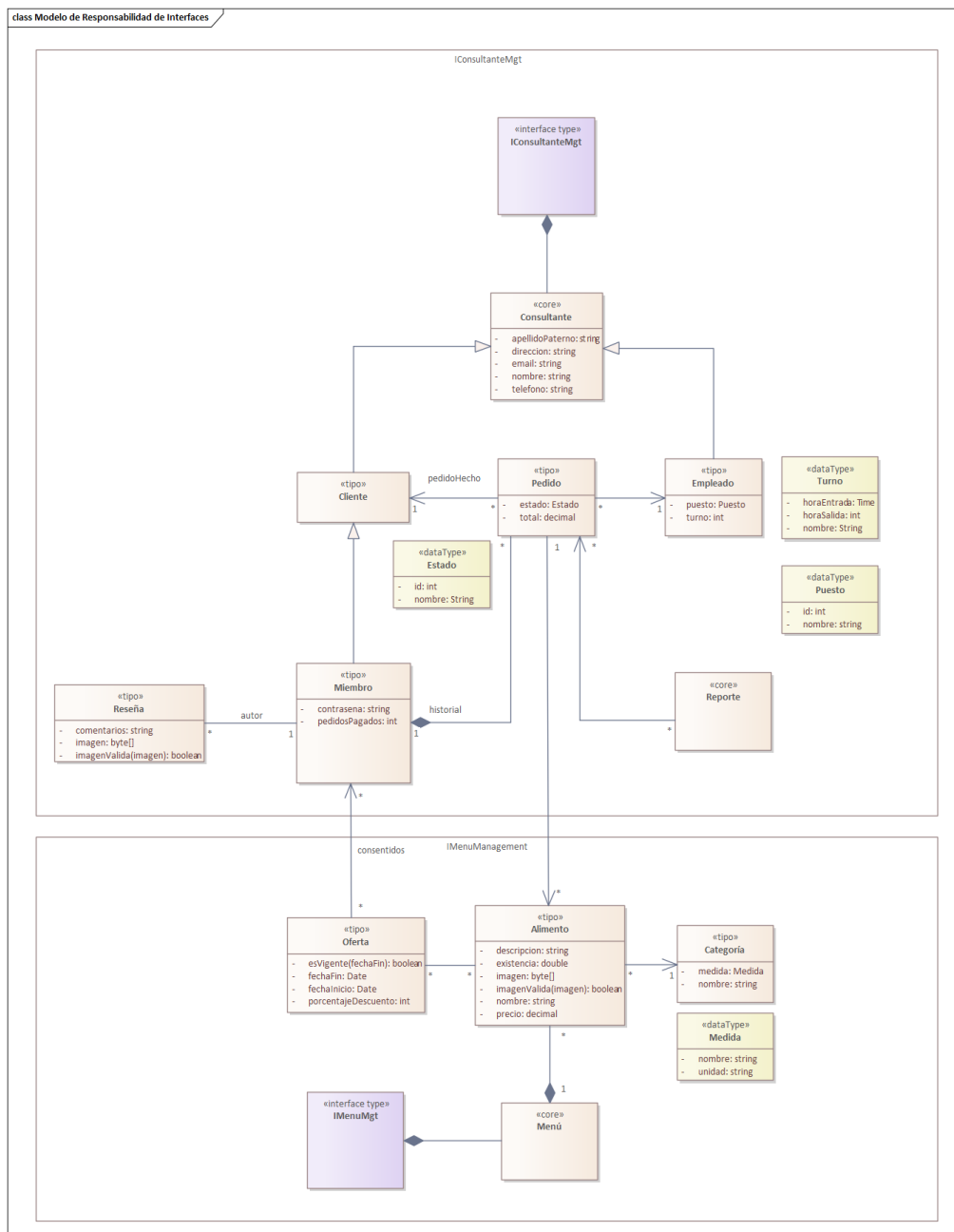


ILUSTRACIÓN 20 DIAGRAMA DE MODELO DE RESPONSABILIDAD DE INTERFACES INICIAL

3.11. Interfaces del Sistema

Por el lado del sistema, se crea una interfaz por cada descripción caso de uso, y se analiza el flujo de ejecución de cada descripción para extraer operaciones pertenecientes a la interfaz de ese caso de uso.

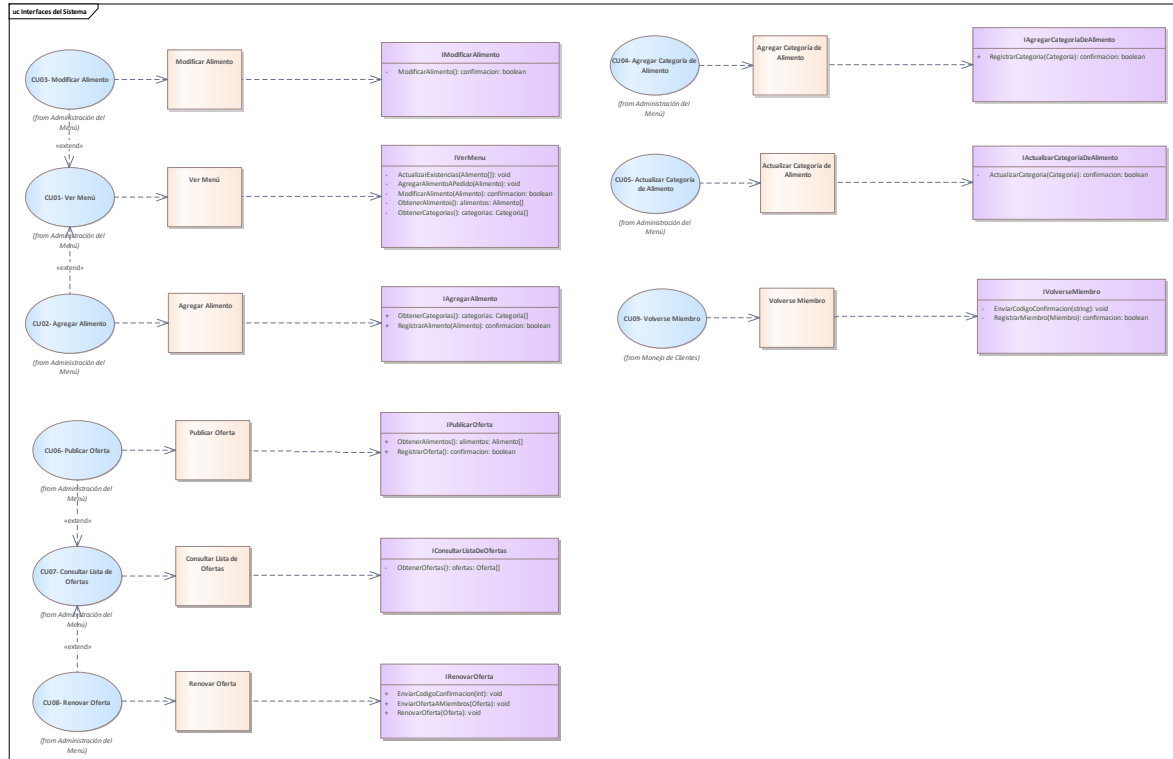


ILUSTRACIÓN 21 MODELO DE INTERFACES DEL SISTEMA

3.12. Especificación de Componentes´

Las interfaces anteriormente descritas son el fundamento para especificar un primer borrador de los componentes necesarios para la construcción de TACOS, mismos que son descritos a continuación.

3.12.1. Especificación de Componentes del Negocio

A continuación, se muestran los dos componentes respectivos de cada interfaz del negocio, descubiertas en la sección 3.2.

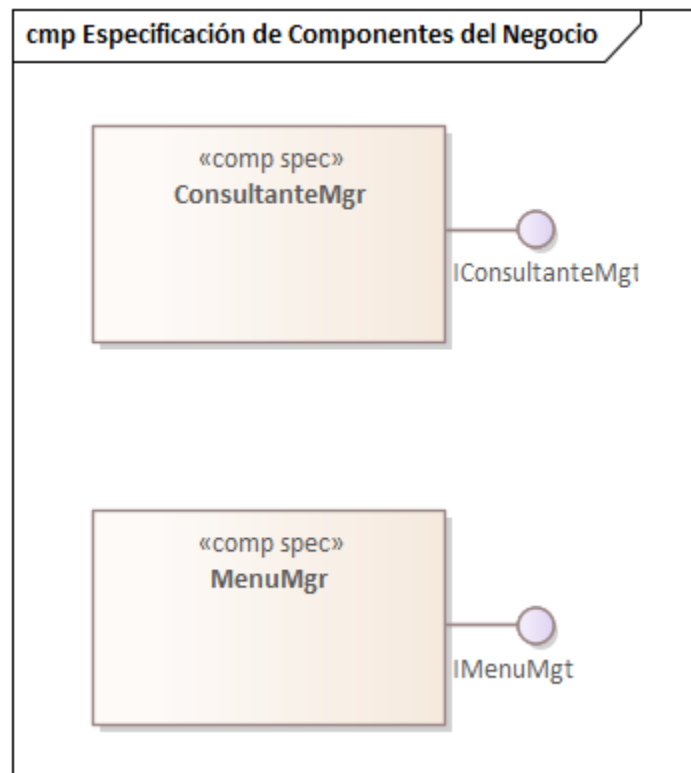


ILUSTRACIÓN 22 ESPECIFICACIÓN DE COMPONENTES DEL NEGOCIO

3.12.2. Especificación de Componentes del Sistema

Con base a las interfaces del sistema (Ilustración 13) se representa el siguiente diagrama como el modelo de especificación de componentes del sistema, en donde se muestra el componente que agrupa las distintas interfaces identificadas para el sistema.

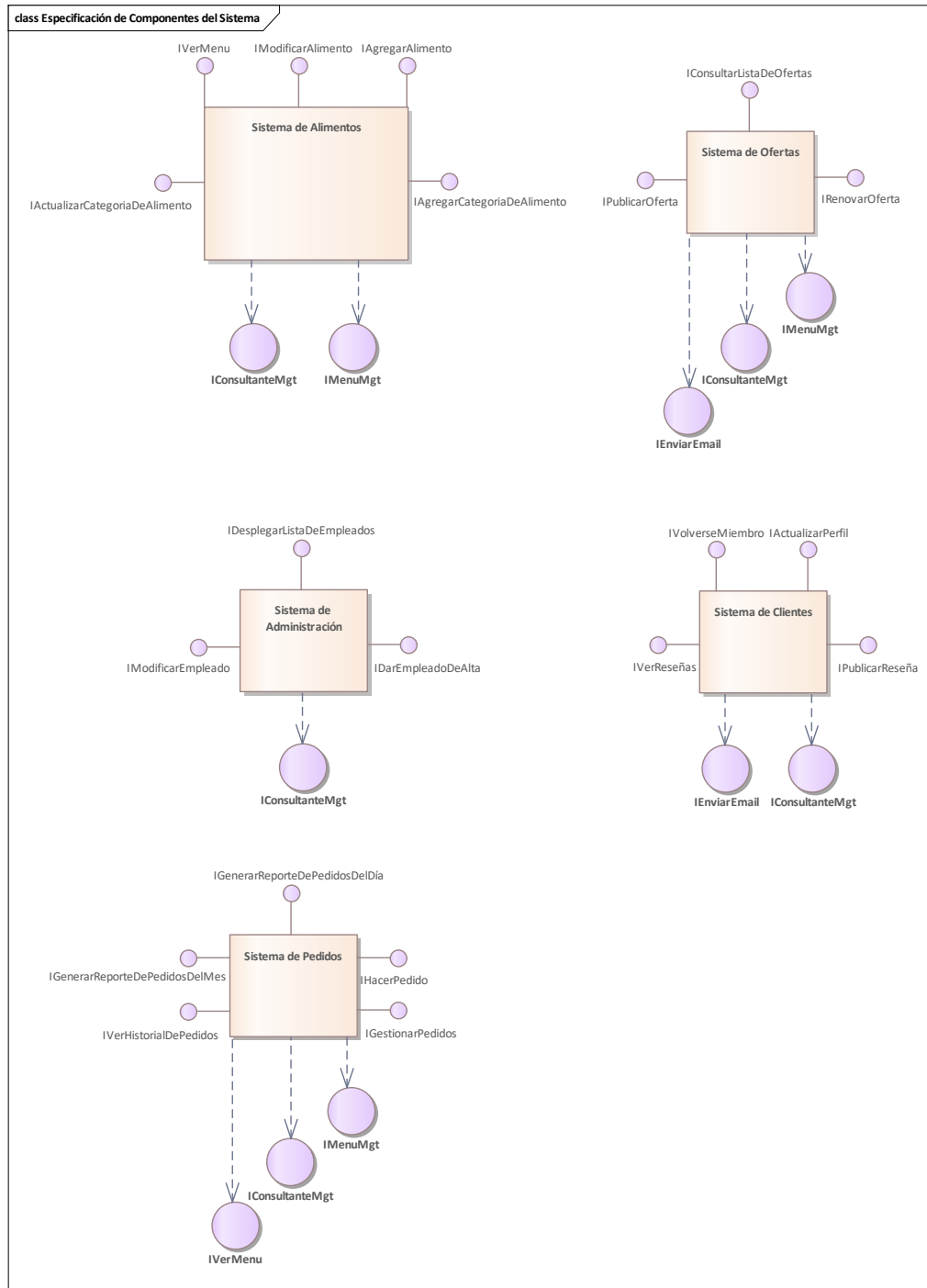


ILUSTRACIÓN 23 ESPECIFICACIÓN DE COMPONENTES DEL SISTEMA

3.12.3. Especificación de Componentes Externos

Para el sistema no solo se utilizarán componentes provenientes de la especificación de requerimientos, sino que también serán ocupados aquellos que contienen interfaces disponibles para uso de terceros tal es el caso del componente “Sistema de Email” que es requerido por otras interfaces del sistema a crear para diversas funcionalidades.

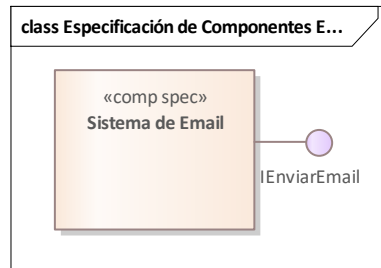


ILUSTRACIÓN 24 ESPECIFICACIÓN DE COMPONENTES EXTERNOS

3.13. Arquitectura de Componentes

El siguiente diagrama se muestran la relación entre los componentes que conforman el sistema T.A.C.O.S, presentando tanto las interfaces requeridas como las expuestas de cada uno y la relación general entre componentes para el correcto funcionamiento del sistema.

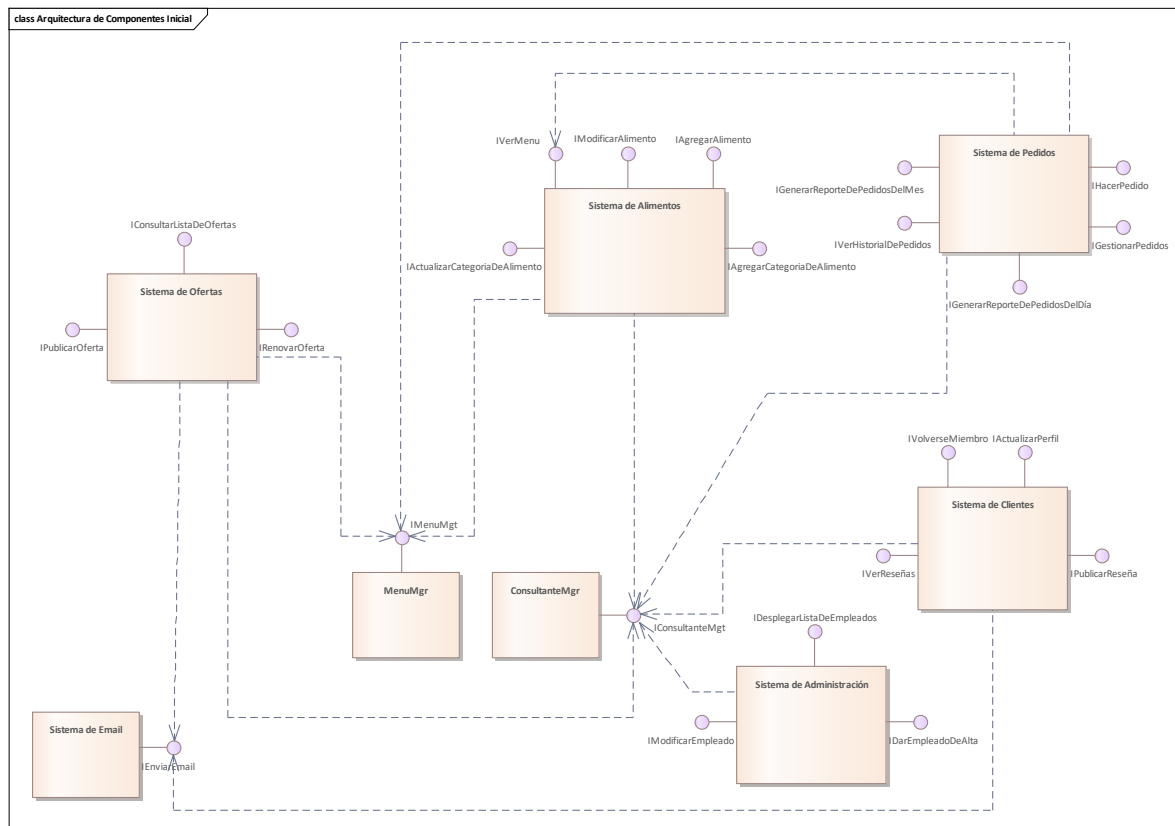


ILUSTRACIÓN 25 ARQUITECTURA DE COMPONENTES T.A.C.O.S

4. Construcción

A continuación, se describen las decisiones, acuerdos y reglas establecidas para la construcción de TACOS.

4.1. Pila tecnológica

En esta sección, se acotan las tecnologías usadas para la construcción del proyecto, y se mencionan los distintos criterios que motivaron cada decisión.

4.1.1. Backend (API)

4.1.1.1. Lenguaje de programación

Se opta por usar C# con el framework ASP.NET, el cual agiliza el desarrollo de servicios web. Para la elección, se consideraron los siguientes criterios:

- Experiencia: El equipo invirtió un aproximado de 5 meses ejercitando la tecnología, desde agosto hasta diciembre del 2022; nuestros conocimientos están frescos.
- Extensibilidad: C#, gracias a la tecnología NuGet, es increíblemente extensible. La integración del repertorio de extensiones con la interfaz gráfica de Visual Studio hace que el proceso de buscar, encontrar y agregar funcionalidad sea muy rápido y sencillo, a diferencia de otras tecnologías que dependen de catálogos externos y el uso de la línea de comandos.
- Tiempo: El corto plazo de desarrollo nos encamina a elegir la tecnología que conocemos en lugar de una que debamos aprender a la par del desarrollo.

4.1.1.2. Tecnología de acceso a datos

Se ha elegido MySQL, aunado con Entity Framework Core (EFC), según los siguientes criterios:

- Experiencia: El equipo lleva toda la carrera trabajando con bases de datos relacionales. Por otro lado, y similar al caso de C#, el equipo ya invirtió múltiples meses en aprender a utilizar LINQ (tecnología de C#) para la consulta de datos en EFC. En aras de agilizar el proceso de construcción, la elección de una base de datos relacional junto a EFC es la más sensata.
- Velocidad de desarrollo: El uso de un ORM ahorra valioso tiempo de desarrollo, al simplificar el proceso de codificar objetos de acceso a datos.
- Apertura: En cuestiones de bases de datos relacionales, MySQL y EFC son compatibles son más tecnologías que otras opciones que el equipo de desarrollo también domina; MS SQL Server y Entity Framework. Esto nos da más flexibilidad a la hora de configurar nuestros entornos de trabajo, e.g. cada miembro del equipo puede utilizar el administrador de bases de datos de su preferencia (Navicat, phpMyAdmin, etc.)

4.1.2. Frontend (cliente de escritorio)

El único elemento a consideración fue el lenguaje de programación; la elección es C#, utilizando Windows Presentation Foundation (WPF), para la construcción de cliente de escritorio rico. El razonamiento es muy similar al de la elección del lenguaje de programación del API (ver 4.1.1.1): el equipo de desarrollo cuenta con cierta experiencia en la tecnología, la tecnología es muy flexible y el corto plazo de entrega nos obliga a aprovechar estas cualidades en la medida de lo posible.

4.2. Estándar de Programación

La intención de desarrollar un documento de estandarización de código consiste en dar el formato o pautas a seguir como herramienta para conseguir el estilo homogéneo y correcto entre los miembros del equipo, además con el uso de este nos permite garantizar a nivel de calidad un código limpio, consistente, legible y comprensible durante todo el desarrollo, buscando el evitando retrasos o reinvertir tiempo en errores de codificación o comprensión de código. Asu vez permite la apropiación de código por parte de o de los programadores dando la aportación de la intención propia del programador ante su código en la implementación de sus propias “buenas prácticas” en un formato consistente y documentado. Por último, el documento se anexa a los contenidos de documentación de código en donde se justifican las practicas implementadas permitiendo un entendimiento mayor al momento de leer el código.

El proyecto se complementa y se rige a su vez por los “C# Coding Convenions”, los cuales definen el estándar base para el desarrollo de código en C# además de las convenciones de .NET.

4.2.1. Reglas de nombrado

- REP01. La referencia a la base de datos puede ser abreviada como “db” o “DB”.
- REP02. Todo nombrado debe ser autodescriptivo a consideración del programador, según el contexto del dominio.
- REP03. Nunca se deben utilizar palabras reservadas.
- REP04. El idioma de código es el español, aunque no se penaliza el uso esporádico de palabras en inglés.

4.2.2. Variables

- REP05. Toda variable debe ser declarada de manera “local” o buscando que su alcance sea el menor posible, para minimizar su alcance dentro del programa, si no colocar un medio correcto para su acceso.
- REP06. Variables de uso local se escribirán en camelCase.
- REP07. Campos privados se escribirán en camelCase. No se antepondrá un guión bajo (_).
- REP08. Campos públicos se escribirán en PascalCase.

Correcto	Incorrecto
<code>String apellidoPaterno = "valor";</code>	<code>String Apellido_Paterno = "valor";</code>

- REP09. Se permite el uso de variables abreviadas o de una sola letra si ésta se usa en un ámbito reducido, e.g. dentro de un predicado LINQ; dentro de un bloque try-catch, etcétera.

```
this.tacosdbContext.Miembros.FirstOrDefault(m => m.IdPersona == persona.Id);
```

4.2.3. Métodos

- REP10. Todo nombre de método debe ser escrito en PascalCase
- REP11. Todo nombre de método debe ser un verbo o una frase verbal. Ejemplos: correr o MandarMensaje, a excepción de métodos booleanos; estos *pueden* usar adjetivos o frases adjetivas: EsValido.
- REP12. Ningún método debe tener más de 5 parámetros.

4.2.4. Clases

- REP13. Los nombres de clase se escriben en PascalCase.
- REP14. Los nombres de clase serán un sustantivo o una frase sustantiva en singular. Ejemplos: Lista o ListaInmutable.

4.2.5. Interfaces

- REP15. Los nombres de interfaces se escriben en PascalCase
- REP16. Los nombres de interfaces tendrán el prefijo.

4.2.6. Componentes gráficos.

- REP17. Los nombres tendrán la estructura: `ClaseDeComponente_NombreDeVariable`.
- REP18. Todas las palabras serán escritas en PascalCase.

4.2.7. Estilo del Código

- REP19. Por cada línea de código no deben de existir más de 200 caracteres, en su defecto utilizar el uso de *Enter* para continuar en la siguiente línea sin aplicar sangría.
- REP20. Ningún salto de línea antes de la llave de apertura.
- REP21. El tabulado preferirá el estilo Allman, pero se deja a consideración del programador.

4.2.8. Comentarios

- REP22. Se permite el uso de comentarios XML para explicar el funcionamiento de elementos públicos del código (clases, métodos, propiedades, etc). Ejemplo:

```

/// <summary>
/// ConfirmarRegistro(). Completa registro de miembro.
/// </summary>
/// <remarks>
Fdsasd

/// Usa el código de confirmación para completar el registro de un miembro.
/// La petición sólo necesita el IdMiembro y el CodigoConfirmacion.
/// </remarks>
/// <response code="200">La petición fue aceptada.</response>
/// <response code="401">El código es incorrecto.</response>
/// <response code="404">No se encontró el miembro solicitado.</response>
/// <response code="500">El servidor falló inesperadamente.</response>
/// <returns>Miembro con el código de confirmación limpio.</returns>
public IActionResult ConfirmarRegistro(Miembro miembro)

```

- REP23. Se permite la supresión de advertencias sobre el comentado XML, siempre y cuando se trate de código cuyo nombre/naturaleza sea obvia en su contexto.

```
public class Staff : IA asociado
{
#pragma warning disable CS1591
public int Id { get; set; }
public string? Contraseña { get; set; }
public int IdPersona { get; set; }
public int IdPuesto { get; set; }
public int IdTurno { get; set; }
```

Para comentarios que no sean XML:

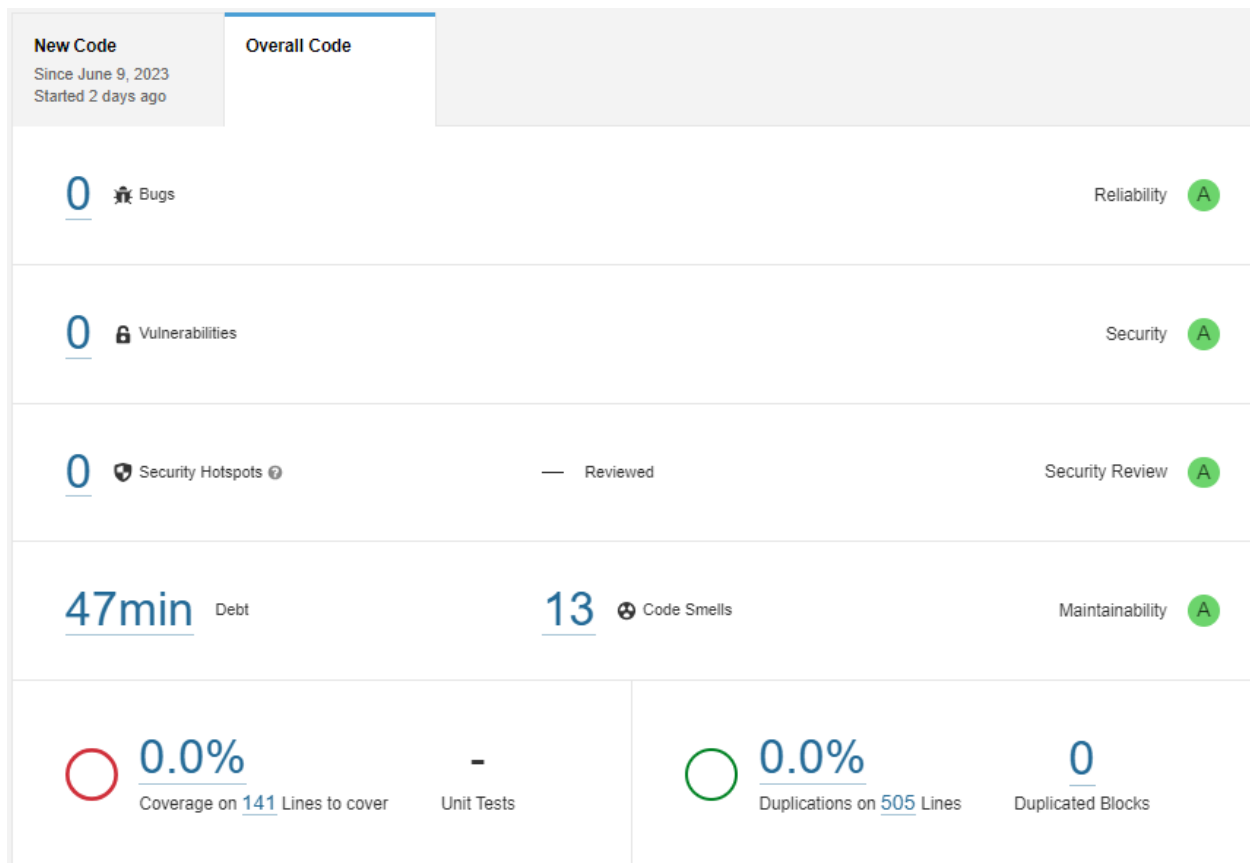
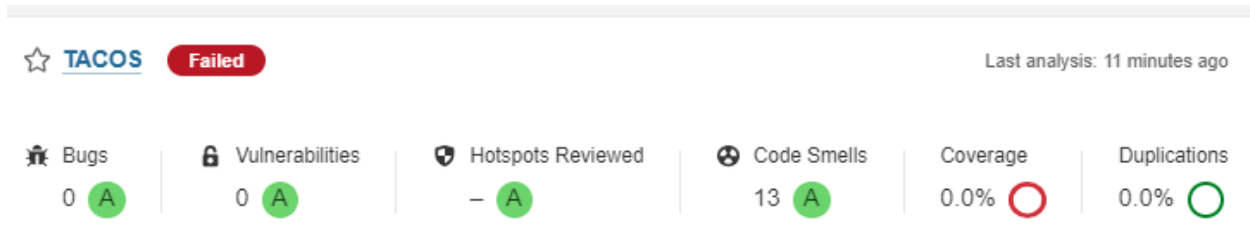
- REP24. Ninguna variable debe ser comentada. Su identificador debe ser suficiente para explicar su propósito.
- REP25. Los comentarios TO DO se permiten, pero estos deben ser removidos del código antes de liberar el software.
- REP26. Los comentarios deben ser serios, pues son parte de la documentación.

Correcto	Incorrecto
//Utilicé un ArrayList porque la clase Vector no tiene los métodos necesarios para hacer esta operación.	/* ##### # la variable "x" significa "edad" xd ##### ##### */

4.3. Análisis Estático

En esta sección, se presentan los resultados de analizar el código fuente del API, mediante la herramienta SonarQube.

En general, se reporta que el código presenta una cantidad marginal de *code smells*, con ningún bug ni vulnerabilidad a la seguridad. La cobertura se reporta en 0% ya que SonarQube requiere el uso de herramientas de terceros, las cuales generen reportes de las pruebas unitarias del proyecto. No se cuenta con estas herramientas, pero las pruebas sí existen (ver sección 5).



TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

☐ Bulk Change

1 / 13 issues 47min effort

TACOS / Controladores/personas/MiembroController.cs

☐ Member 'EnviarCodigoConfirmacion' does not access instance data and can be marked as static ROSLYN 2 days ago L81 No tags

☐ Code Smell Minor Open Not assigned 0min effort Comment

TACOS / Modelos/MiembroValidador.cs

☐ Use 'GeneratedRegexAttribute' to generate the regular expression implementation at compile-time. ROSLYN 2 days ago L48 No tags

☐ Code Smell Info Open Not assigned 0min effort Comment

☐ Use 'GeneratedRegexAttribute' to generate the regular expression implementation at compile-time. ROSLYN 2 days ago L49 No tags

☐ Code Smell Info Open Not assigned 0min effort Comment

☐ Use 'GeneratedRegexAttribute' to generate the regular expression implementation at compile-time. ROSLYN 2 days ago L53 No tags

☐ Code Smell Info Open Not assigned 0min effort Comment

☐ Use 'GeneratedRegexAttribute' to generate the regular expression implementation at compile-time. ROSLYN 2 days ago L54 No tags

☐ Code Smell Info Open Not assigned 0min effort Comment

TACOS / Negocio/ConsultanteMgr.cs

☐ Dereference of a possibly null reference. ROSLYN 57 minutes ago L47 No tags

☐ Code Smell Major Open Not assigned 0min effort Comment

☐ Converting null literal or possible null value to non-nullable type. ROSLYN 57 minutes ago L65 No tags

☐ Code Smell Major Open Not assigned 0min effort Comment

☐ Converting null literal or possible null value to non-nullable type. ROSLYN 57 minutes ago L70 No tags

☐ Code Smell Major Open Not assigned 0min effort Comment

☐ The result of the expression is always 'false' since a value of type 'int' is never equal to 'null' of type 'int?' ROSLYN 57 minutes ago L322 No tags

☐ Code Smell Major Open Not assigned 0min effort Comment

TACOS / TACOSMenuAPI/Controladores/MenuController.cs

☐ Remove this unread private field 'logger' or refactor the code to use its value. 2 days ago L19 cwe, unused

☐ Code Smell Major Open Not assigned 5min effort Comment

TACOS / TACOSMenuAPI/Modelos/TacosdbContext.cs

☐ Supply an implementation for the partial method, otherwise this call will be ignored. 2 days ago L83 suspicious

☐ Code Smell Minor Open Not assigned 20min effort Comment

☐ Supply an implementation for this partial method. 2 days ago L86 suspicious

☐ Code Smell Minor Open Not assigned 20min effort Comment

TACOS / TACOSMenuAPI/Program.cs

☐ Remove this empty statement. 2 days ago L14 unused

☐ Code Smell Minor Open Not assigned 2min effort Comment

13 of 13 shown

4.4. Buenas Prácticas de Programación Implementadas

A continuación, se enumeran las buenas prácticas de programación implementadas en el proyecto, según se nos han inculcado a lo largo de nuestros estudios de Ingeniería de Software.

4.4.1. Nombramiento

Las variables, clases, propiedades y métodos han sido nombrados con la intención de acercar el código al español cotidiano en la medida de lo posible. Esto incrementa la legibilidad y, por ende, la mantenibilidad y extensibilidad del código.

```

2 references
public Respuesta<Pedido> RegistrarPedido(Pedido nuevoPedido)
{
    if (nuevoPedido.Alimentospedidos.IsNullOrEmpty())
    {
        return new Respuesta<Pedido> {Codigo = 400, Mensaje = Mensajes.RegistrarPedido_400 };
    }
    foreach (Alimentospedido alimento in nuevoPedido.Alimentospedidos)
    {
        if (alimento.Cantidad < 1)
        {
            return new Respuesta<Pedido> {Codigo = 400, Mensaje = Mensajes.RegistrarPedido_400 };
        }
    }
    this.tacosdbContext.Pedidos.Add(nuevoPedido);
    try
    {
        this.tacosdbContext.SaveChanges();
    }
    catch (DbUpdateException)
    {
        return new Respuesta<Pedido> {Codigo = 500, Mensaje = Mensajes.ErrorInterno };
    }
    return new Respuesta<Pedido> {Codigo = 200, Mensaje = Mensajes.OperacionExitosa, Datos = nuevoPedido };
}

```

4.4.2. Comentarios

En acorde con lo definido en el estándar (ver RP24), los comentarios no-XML deben ser restringidos: sólo deben ser usados para explicar el *porqué* del código; el *qué* debe ser explicado por los nombres usados y por los comentarios XML. Siguiendo este criterio, se cuentan con métodos que explican decisiones en el código que podrían parecer poco ortodoxas, para facilitar su comprensión.

```

IAsociado asociadoEncontrado;
if (credenciales.EsStaff)
{
    //El Staff/Miembro podría ser nulo; esta validación se hace en el
    //siguiente if (asociadoEncontrado == null), para no tener que
    //validar dos veces.
    respuesta.Staff = personas.First().Staff.FirstOrDefault();
    asociadoEncontrado = respuesta.Staff;
}
else
{
    respuesta.Miembro = personas.First().Miembros.FirstOrDefault();
    asociadoEncontrado = respuesta.Miembro;
}
if (asociadoEncontrado == null)
{
    return new RespuestaCredenciales {Codigo = 401, Mensaje = Mensajes.IniciarSesion_401 };
}

```

4.4.3. Portabilidad

Se debe evitar el uso de valores *hardcodeados* en la medida de lo posible. Esto fomenta atributos de calidad como la facilidad de mantenimiento, facilidad de modificación, modularidad y reutilización del código. Ejemplos de esta práctica en el código son los archivos appsettings.json, los cuales contienen información que puede ser fácilmente modificada, tal como la cadena de conexión a la base de datos.

The screenshot shows two files in Visual Studio. The top file is `appsettings.json` with the following content:

```

1 {
2   "Logging": {
3     "LogLevel": {
4       "Default": "Information",
5       "Microsoft.AspNetCore": "Warning"
6     }
7   },
8   "ConnectionStrings": {
9     "tacosdb": "server=localhost;database=tacosdb;uid=tacosUser;pwd=T4C05"
10  },
11   "AllowedHosts": "*"
12 }
13
14

```

The bottom file is `Program.cs` with the following content:

```

27 builder.Services.AddDbContext<TacosdbContext>(options =>
28     options.UseMySQL(
29         builder.Configuration.GetConnectionString("tacosdb"),
30         Microsoft.EntityFrameworkCore.ServerVersion.Parse("8.0.25-mysql")
31     ));

```

Otro ejemplo de esto, es la utilización de los archivos `.RESX`, los cuales contienen diccionarios de valores reutilizables. En el proyecto, se utilizan para enviar retroalimentación al cliente.

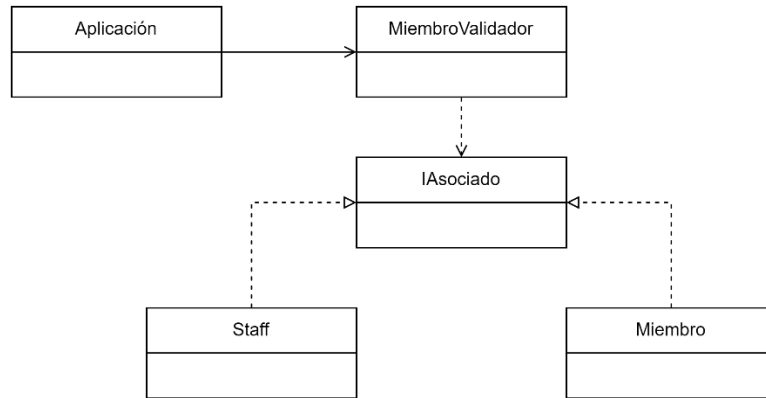
The screenshot shows the `Mensajes.resx` resource file in Visual Studio. It contains a table of localized messages:

Name	Value
ActualizarPedido_400	Debe seleccionar un pedido.
ActualizarPedido_403	El pedido no puede cambiar su estado, pues ya fue pagado.
ActualizarPedido_404	El pedido solicitado no existe.
ActualizarPedido_422	El pedido solicitado pertenece a un miembro distinto.
BorrarResena_400	Se requiere un ID de reseña válido para eliminar el registro.
BorrarResena_404	Ningun registro coincide con la reseña que desea eliminar.
ConfirmarRegistro_401	El código es incorrecto.
ConfirmarRegistro_404	No se encontró el miembro solicitado.
ErrorInterno	Error de servidor, intente de nuevo más tarde.

4.4.4. Patrones de diseño y Reutilización

Los patrones de diseño sirven para agilizar el proceso de implementación de un sistema, pues proveen de soluciones probadas para problemas recurrentes. En nuestro caso, debemos validar registro de personas; y siendo que tenemos dos tipos de usuarios registrados — Miembro y Staff — los cuales comparten la mayoría de sus atributos, no tendría sentido crear mecanismos de validación únicos para cada clase. Por esta razón, se implementó el patrón *Servant*, el cual permite ofrecer funcionalidad a múltiples clases sin tener que definir dicha funcionalidad por separado. Esto, a su vez, fomenta la reutilización de código, pues la validación se puede ofrecer a cualquier clase siempre y cuando implementen la misma interfaz.

TACO API COMMERCIAL ORDER SYSTEM (TACOS) E.R.S



Otro patrón utilizado en este proyecto es la *inyección de dependencias* (*Dependency Injection*) el cual es un patrón utilizado en muchas plataformas de software.

Este patrón permite reducir el acoplamiento entre componentes y mejorar la modularidad, la facilidad de prueba del código y Flexibilidad que como fue explicada anteriormente permite que la configuración de las dependencias se puede realizar externamente, lo que permite una mayor flexibilidad en la composición de componentes y la adaptación del sistema a diferentes escenarios.

Para el caso de este proyecto este patrón es ampliamente utilizado. Es visible el archivo **Program.cs** de cada proyecto.

```
var builder = WebApplication.CreateBuilder(args);

builder.Services.AddJwtAuthentication();
builder.Services.AddControllers()
    .AddJsonOptions(x =>
        x.JsonSerializerOptions.ReferenceHandler = ReferenceHandler.Preserve);

builder.Services.AddEndpointsApiExplorer();
builder.Services.AddSwaggerGen(c =>
{
    var xmlFile = $"{Assembly.GetExecutingAssembly().GetName().Name}.xml";
    var xmlPath = Path.Combine(AppContext.BaseDirectory, xmlFile);
    c.IncludeXmlComments(xmlPath);
});

builder.Services.AddDbContext<TacosdbContext>(options =>
{
    options.UseMySQL(
        builder.Configuration.GetConnectionString("tacosdb"),
        Microsoft.EntityFrameworkCore.ServerVersion.Parse("8.0.25-mysql")
    );
});
builder.Services.AddScoped<IConsultanteMgt, ConsultanteMgr>();

var app = builder.Build();

if (app.Environment.IsDevelopment())
{
    app.UseSwagger();
    app.UseSwaggerUI();
}
```

Y es utilizado en la mayoría de las clases que necesitan las dependencias que fueron cargadas.

```
0 references
public StaffController(ILogger<StaffController> logger,
    IConsultanteMgt consultanteMgr)
{
    this.logger = logger;
    _consultanteMgr = consultanteMgr;
}
```

4.4.5. Implementación de interfaces externas o existentes

Para el caso de aquellas funcionalidades que requieren implementación de acciones conocidas como enviar un email, el equipo de desarrollo opto por tomar interfaces existentes para no verse en la necesidad de crear la funcionalidad desde cero sino poder hacer uso de una existente, lo anterior fue contemplado desde fases iniciales del proyecto en los puntos 3.12 (Especificación de componentes) y 3.13 (Arquitectura de componentes) de este documento.

```
using System.Linq;
using System.Net;
using System.Net.Mail;
using System.Runtime.Serialization;
```

```
string remitente = Properties.CredencialesEmail.Direccion;
string senderPassword = Properties.CredencialesEmail.Contrasena;
MailMessage mensaje = new MailMessage();
mensaje.From = new MailAddress(remitente);
mensaje.Subject = email.Asunto;
```

4.4.6. Documentación del API

Debido a la naturaleza del framework de desarrollo y el lengua de programación especificados en el punto 4.1 (Pila tecnológica) de este documento el archivo que proporciona una implementación en OPENAPI tenía incompatibilidades con el medio de desarrollo empleado por lo que se buscó la alternativa que mejor se adaptara, en este caso el uso de **comentarios XML**, medio por el cual Swagger permite documentar una API.

```
/// <summary>
/// BorrarResena(int idResena). Borra una reseña seleccionada mediante su Id.
/// </summary>
/// <remarks>
/// Solo es necesario el Id de la reseña para el proceso.
/// </remarks>
/// <response code="204">
/// La operación fue exitosa.
/// </response>
/// <response code="400">Se requiere un ID de reseña válido para eliminar el registro.</response>
/// <response code="404">Ningun registro coincide con la reseña que desea eliminar.</response>
/// <response code="500">Error de servidor, intente de nuevo más tarde.</response>
/// <returns>Resultado sin contenido</returns>
[ProducesResponseType(typeof(NoContentResult), StatusCodes.Status204NoContent)]
[ProducesResponseType(typeof(Error), StatusCodes.Status400BadRequest)]
[ProducesResponseType(typeof(Error), StatusCodes.Status404NotFound)]
[ProducesResponseType(typeof(Error), StatusCodes.Status500InternalServerError)]
[Produces("application/json")]
```

```
[HttpPost(Name = "BorrarResena")]
[Authorize]
```

0 references

```
public IActionResult BorrarResena(int idResena)
{
    try
    {
        this.consultanteMgr.BorrarResena(idResena);
        return new NoContentResult();
    }
    catch (HttpRequestException httpRequestException)
    {
        Dictionary<string, string> respuestasExcepcion = new Dictionary<string, string>();
        respuestasExcepcion.Add("400", Mensajes.BorrarResena_400);
```

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

Por este medio podemos proporcionar información adicional y descripciones detalladas sobre el endpoint, sus parámetros, respuestas y otras consideraciones relevantes.

Lo anterior tendrá un resultado visual como el que se muestra a continuación.

Resenas

GET /Resenas ObtenerResenas(). Regresa una lista con todos las reseñas.

POST /Resenas BorrarResena(int idResena). Borra una reseña seleccionada mediante su id.

Solo es necesario el Id de la reseña para el proceso.

Parameters

Name	Description
idResena	<input type="text" value="idResena"/>

Try it out

Responses

Code	Description	Links
204	La operación fue exitosa. <div><div>Media type</div><div>application/json</div><div>Controls Accept header.</div><div>Example Value Schema</div><div>{</div></div>	No links
400	Se requiere un ID de reseña válido para eliminar el registro. <div><div>Media type</div><div>application/json</div><div>Example Value Schema</div><div>{</div></div>	No links
404	Ningun registro coincide con la reseña que desea eliminar. <div><div>Media type</div><div>application/json</div><div>Example Value Schema</div><div>{</div></div>	No links
500	Error de servidor, intente de nuevo más tarde. <div><div>Media type</div><div>application/json</div><div>Example Value Schema</div><div>{</div></div>	No links

5. Pruebas

Posterior a la construcción de los servicios y del cliente, estos fueron probados para validar su correcto funcionamiento. El proceso de pruebas seguido por Darkmode se divide en dos partes:

1. Pruebas unitarias: realizadas desde el API, donde se prueban los endpoints y su lógica del negocio de forma directa.
2. Pruebas de integración: hechas en el cliente, donde se prueba el envío de peticiones desde el cliente; el enrutamiento del Gateway; el funcionamiento de los controladores (o del servidor gRPC, en su caso); la lógica del negocio de las tres APIs; y el procesamiento de la respuesta recibida en el cliente. Estas pruebas se basan en los casos diseñados y presentados a continuación.

El enfoque de los casos de prueba es similar a la técnica *camino de ejecución de casos de uso*, en el que cada caso de prueba valida el funcionamiento de un flujo del caso de uso en cuestión: el normal, los alternos y los de excepción. De esta manera, los casos de prueba a continuación descritos, buscan probar el funcionamiento de los servicios expuestos por las APIs — tanto en flujos normales, como en flujos alternos.

5.1. Casos de Prueba

5.1.1. Login

Sistema: Cliente de escritorio.	Servicio: IniciarSesion
--	--------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-1	Probar el flujo normal.	Email: admin Contraseña: ASDFasdf1234 EsStaff: false	Código: 200. Mensaje: Operación exitosa.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 200. Mensaje: Operación exitosa.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: IniciarSesion
--	--------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-2	Probar el flujo de excepción, cuando se encuentra el usuario, pero su contraseña en la base de datos no	Email: string Contraseña: string EsStaff: false	Código: 500. Mensaje: Error interno.

	fue encriptada correctamente.		
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 500. Mensaje: Error interno.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: IniciarSesion
--	--------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-3	Probar el flujo alterno, cuando el Email está vacío.	Email: Contrasena: string EsStaff: false	Código: 400. Mensaje: Todos los campos son obligatorios.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 400. Mensaje: Todos los campos son obligatorios.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: IniciarSesion
--	--------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-4	Probar el flujo alterno, cuando la Contraseña está vacía.	Email: admin Contrasena: EsStaff: false	Código: 400. Mensaje: Todos los campos son obligatorios.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 400. Mensaje: Todos los campos son obligatorios.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: IniciarSesion
--	--------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-5	Probar el flujo alterno, cuando todo está vacío.	Email: Contrasena: EsStaff: false	Código: 400. Mensaje: Todos los campos son obligatorios.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 400. Mensaje: Todos los campos son obligatorios.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: IniciarSesion
--	--------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-6	Probar el flujo alterno, cuando el Email no existe en la base de datos.	Email: 1234reqwwer45 Contrasena: fdsasdf EsStaff: false	Código: 401. Mensaje: No se encontró ninguna cuenta con ese email y/o contraseña.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 401. Mensaje: No se encontró ninguna cuenta con ese email y/o contraseña.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: IniciarSesion
--	--------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-7	Probar el flujo alterno, cuando el Email existe, pero la contraseña es incorrecta.	Email: admin Contrasena: asdf EsStaff: false	Código: 401. Mensaje: No se encontró ninguna cuenta con ese email y/o contraseña.
Resultado del caso de prueba	Prueba exitosa	Fecha	

Código: 401. Mensaje: No se encontró ninguna cuenta con ese email y/o contraseña.	Sí.	10/06/2023	
--	-----	------------	--

Total de pruebas realizadas: 7	Cobertura: 100%
---------------------------------------	------------------------

5.1.2. Miembro

Sistema: Cliente de escritorio.	Servicio: RegistrarMiembro
--	-----------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-8	Probar el flujo normal.	Contrasena: Asdf1234 Nombre: PRUEBA ApellidoPaterno:PRUEBA ApellidoMaterno:PRUEBA Direccion:PRUEBA Email: {aleatorio}@{aleatorio}.com Telefono:2288184812	Código: 200. Mensaje: Operación exitosa.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 200. Mensaje: Operación exitosa	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: RegistrarMiembro
--	-----------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-9	Probar el flujo alterno, cuando el Email es incorrecto.	Contrasena: Asdf1234 Nombre: PRUEBA ApellidoPaterno:PRUEBA ApellidoMaterno:PRUEBA Direccion:PRUEBA	Código: 400. Mensaje: El email no tiene el formato correcto.

		Email:asdf Telefono:2288184812	
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 400. Mensaje: El email no tiene el formato correcto.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: RegistrarMiembro
--	-----------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-10	Probar el flujo alterno, cuando la contraseña no sigue la política.	Contraseña: asdf Nombre: PRUEBA ApellidoPaterno:PRUEBA ApellidoMaterno:PRUEBA Direccion:PRUEBA Email: {aleatorio}@{aleatorio}.com Telefono:2288184812	Código: 400. Mensaje: La contraseña debe tener al menos 8 caracteres: al menos una letra minúscula; al menos una mayúscula y al menos un número
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 400. Mensaje: La contraseña debe tener al menos 8 caracteres: al menos una letra minúscula; al menos una mayúscula y al menos un número	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: RegistrarMiembro
--	-----------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-11	Probar el flujo alterno, cuando el nombre no es alfabético.	Contrasena: Asdf1234 Nombre: --DROP DATABASE ApellidoPaterno:PRUEBA ApellidoMaterno:PRUEBA Direccion:PRUEBA Email: {aleatorio}@{aleatorio}.com Telefono:2288184812	Código: 400. Mensaje: Los nombres sólo pueden contener letras.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 400. Mensaje: Los nombres sólo pueden contener letras.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: RegistrarMiembro
--	-----------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-12	Probar el flujo alterno, cuando el apellido paterno no es alfabético.	Contrasena: Asdf1234 Nombre: PRUEBA ApellidoPaterno: --DROP DATABASE ApellidoMaterno:PRUEBA Direccion:PRUEBA Email: {aleatorio}@{aleatorio}.com Telefono:2288184812	Código: 400. Mensaje: Los nombres sólo pueden contener letras.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 400. Mensaje: Los nombres sólo	Sí.	10/06/2023	

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

pueden contener letras.			
-------------------------	--	--	--

Sistema: Cliente de escritorio.	Servicio: RegistrarMiembro
--	-----------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-13	Probar el flujo alterno, cuando el apellido materno no es alfabético.	Contrasena: Asdf1234 Nombre: PRUEBA ApellidoPaterno: PRUEBA ApellidoMaterno: --DROP DATABASE Direccion:PRUEBA Email: {aleatorio}@{aleatorio}.com Telefono:2288184812	Código: 400. Mensaje: Los nombres sólo pueden contener letras.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 400. Mensaje: Los nombres sólo pueden contener letras.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: RegistrarMiembro
--	-----------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-14	Probar el flujo alterno, cuando el teléfono es inválido.	Contrasena: Asdf1234 Nombre: PRUEBA ApellidoPaterno: PRUEBA ApellidoMaterno: PRUEBA Direccion:PRUEBA Email: {aleatorio}@{aleatorio}.com	Código: 400. Mensaje: Los nombres sólo pueden contener letras.



		Telefono: --DROP DATABASE	
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 400. Mensaje: Los nombres sólo pueden contener letras.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ConfirmarRegistro
--	------------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-15	Probar el flujo normal.	La prueba registra a un miembro con las mismas entradas que en TEST-8; posterior al registro obtiene el IdMiembro y CódigoConfirmacion del Miembro que acaba de registrar, mismos que usa para probar el servicio; las entradas IdMiembro y CódigoConfirmacion cambian en cada ejecución de la prueba.	Código: 200. Mensaje: Operación exitosa.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 200. Mensaje: Operación exitosa.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ConfirmarRegistro
--	------------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-16	Probar el flujo normal.	La prueba registra a un miembro con las mismas entradas que en TEST-8; posterior al registro obtiene el IdMiembro y cambia el CodigoConfirmacion por uno erróneo: 420.	Código: 401. Mensaje: El código es incorrecto.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 401. Mensaje: El código es incorrecto.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ConfirmarRegistro
--	------------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-17	Probar el flujo alterno, cuando el código es incorrecto.	La prueba registra a un miembro con las mismas entradas que en TEST-8; posterior al registro obtiene el IdMiembro y cambia el CodigoConfirmacion por uno erróneo: 420.	Código: 401. Mensaje: El código es incorrecto.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 401. Mensaje: El código es incorrecto.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ConfirmarRegistro
--	------------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-18	Probar el flujo alterno, cuando el Miembro no existe.	La prueba registra a un miembro con las mismas entradas que en TEST-8; posterior al registro cambia el IdMiembro que acaba de obtener por uno erróneo: 654654.	Código: 404. Mensaje: No se encontró el miembro solicitado.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 404. Mensaje: No se encontró el miembro solicitado.	Sí.	10/06/2023	

Total de pruebas realizadas: 10	Cobertura: 100%
--	------------------------

5.1.3. Menu

Sistema: Cliente de escritorio.	Servicio: ObtenerMenu
--	------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-19	Probar el flujo normal.	N/A	Código: 200. Menú con 2 alimentos.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 200. Menú con 2 alimentos.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ActualizarExistenciaAlimentos
--	--

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-20	Probar el flujo normal.	IdAlimento: 2 Cantidad: -1	Código: 200. La existencia del IdAlimento = 2 decreció en 1.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 200. La existencia del IdAlimento = 2 decreció en 1.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ActualizarExistenciaAlimentos
--	--

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-21	Probar el flujo alternativo, cuando el IdAlimento no existe.	IdAlimento: -1 Cantidad: -1	Código: 404. Mensaje: El alimento solicitado no existe.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 404. Mensaje: El alimento solicitado no existe.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ActualizarExistenciaAlimentos
--	--

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-22	Probar el flujo alterno, cuando el IdAlimento no existe.	IdAlimento: -1 Cantidad: -1	Código: 404. Mensaje: El alimento solicitado no existe.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 404. Mensaje: El alimento solicitado no existe..	Sí.	10/06/2023	

Total de pruebas realizadas: 4	Cobertura: 66%
---------------------------------------	-----------------------

5.1.4. Pedido

Sistema: Cliente de escritorio.	Servicio: RegistrarPedido
--	----------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-23	Probar el flujo normal.	Pedido con dos alimentos: - IdAlimento: 1, Cantidad 420 - IdAlimento 2, Cantidad: 69	Código: 200. Mensaje: Operación exitosa.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 200. Mensaje: Operación exitosa.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: RegistrarPedido
--	----------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-24	Probar el flujo alterno, cuando el pedido no tiene alimentos.	Pedido vacío.	Mensaje: El pedido debe contener al menos un alimento, y todos los alimentos deben tener una cantidad mayor a 0.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Mensaje: El pedido debe contener al menos un alimento, y todos los alimentos deben tener una cantidad mayor a 0.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ActualizarPedido
--	-----------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-25	Probar el flujo normal.	IdPedido: 13 Estado: 0	Código: 200.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 200.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ActualizarPedido
--	-----------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-26	Probar el flujo alternativo, cuando el IdPedido = 0.	IdPedido: 0 IdMiembro: 20 Estado: 0	Mensaje: Debe seleccionar un pedido.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Mensaje: Debe seleccionar un pedido.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ActualizarPedido
--	-----------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-27	Probar el flujo alternativo, el Pedido le pertenece a un miembro distinto.	IdPedido: 13 IdMiembro: -1 Estado: 0	Mensaje: El pedido solicitado pertenece a un miembro distinto.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Mensaje: El pedido solicitado pertenece a un miembro distinto.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ActualizarPedido
--	--

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-28	Probar el flujo alterno, el Pedido solicitado no existe.	IdPedido: 42069 IdMiembro: 20 Estado: 0	Mensaje: El pedido solicitado no existe.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Mensaje: El pedido solicitado no existe.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ActualizarPedido
--	--

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-29	Probar el flujo alterno, el Pedido solicitado no existe.	IdPedido: 42069 IdMiembro: 20 Estado: 0	Mensaje: El pedido solicitado no existe.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Mensaje: El pedido solicitado no existe.	Sí.	10/06/2023	

Sistema: Cliente de escritorio.	Servicio: ObtenerPedidosEntre
--	---

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-30	Probar el flujo normal.	Entre: 10/05/2023 Hasta: 11/05/2023	Código: 200 Numero de pedidos: 7 Todos los pedidos entran en el rango especificado.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 200	Sí.	10/06/2023	

TACO API COMMERCIAL ORDER SYSTEM (TACOS)

E.R.S

Numero de pedidos: 7 Todos los pedidos entran en el rango especificado.			
---	--	--	--

Sistema: Cliente de escritorio.	Servicio: ObtenerPedidosEntre
--	---

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-31	Probar el flujo alterno, cuando no existen pedidos en el rango de fechas especificado.	Entre: 11/05/2023 Hasta: 10/05/2023	Mensaje: No se encontraron pedidos en el rango especificado.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Mensaje: No se encontraron pedidos en el rango especificado.	Sí.	10/06/2023	

Total de pruebas realizadas: 10	Cobertura: 100%
--	-------------------------------

5.1.5. Puestos

Sistema: Cliente de escritorio.	Servicio: ObtenerPuestos
--	--

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-32	Probar el flujo normal para obtener los puestos existentes.	N/A.	Lista de Puestos.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Lista de Puestos.	Sí.	11/06/2023	

Total de pruebas realizadas: 1	Cobertura: 100%
---------------------------------------	-------------------------------



5.1.6. Turnos

Sistema: Cliente de escritorio.	Servicio: ObtenerTurnos
--	---------------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-33	Probar el flujo normal para obtener los turnos existentes.	N/A.	Lista de Turnos.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Lista de Turnos.	Sí.	11/06/2023	

Total de pruebas realizadas: 1	Cobertura: 100%
---------------------------------------	-------------------------------

5.1.7. Reseñas

Sistema: Cliente de escritorio.	Servicio: ObtenerReseñas
--	--

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-34	Probar el flujo normal para obtener las reseñas existentes.	N/A.	Lista de Reseñas.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Lista de Reseñas.	Sí.	11/06/2023	

Total de pruebas realizadas: 1	Cobertura: 100%
---------------------------------------	-------------------------------

Sistema: Cliente de escritorio.	Servicio: BorrarReseña
--	--------------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-35	Probar el flujo normal para borrar una reseña.	idReseña = 18 (Cualquier otro existente en el sistema)	Código: 204

Resultado del caso de prueba	Prueba exitosa	Fecha	
Código 204	Sí.	11/06/2023	

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-36	Probar el flujo alternativo para borrar una reseña mediante un idReseña que no existe en el sistema.	idReseña = -99997 (Cualquier otro no existente en el sistema)	Código: 404 Mensaje: Ningún registro coincide con la reseña que desea eliminar.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 404 Mensaje: Ningún registro coincide con la reseña que desea eliminar.	Sí.	11/06/2023	

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-37	Probar el flujo alternativo para borrar una reseña mediante un idReseña que no es válido.	idReseña = 0 (o null)	Código: 400 Mensaje: Se requiere un ID de reseña válido para eliminar el registro.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código: 400 Mensaje: Se requiere un ID de reseña válido para eliminar el registro.	Sí.	11/06/2023	

Total de pruebas realizadas: 3	Cobertura: 100%
---------------------------------------	------------------------

5.1.8. Staff

Sistema: Cliente de escritorio.	Servicio: RegistrarStaff
--	---------------------------------

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-38	Probar el flujo normal para registrar a una nueva persona en el staff.	Nombre = "Test", ApellidoMaterno = "Test", ApellidoPaterno = "Test", Direccion = "Calle Test #Tes Col.Test", Email = test@gmail.com , Telefono = "2244556677", IdTurno = 1, IdPuesto = 3, Contrasena = "asdASD123*asads".	Código 204
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código 204	Sí.	11/06/2023	

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-39	Probar el flujo alternativo para registrar a una nueva persona en el staff con el nombre inválido.	Nombre = "Test****[--1-", ApellidoMaterno = "Test", ApellidoPaterno = "Test", Direccion = "Calle Test #Tes Col.Test", Email = test@gmail.com , Telefono = "2244556677", IdTurno = 1, IdPuesto = 3, Contrasena = "asdASD123*asads".	Código 400 Mensaje: Los nombres sólo pueden contener letras.

Resultado del caso de prueba	Prueba exitosa	Fecha	
Código 400 Mensaje: Los nombres sólo pueden contener letras.	Sí.	11/06/2023	

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-40	Probar el flujo alternativo para registrar a una nueva persona en el staff con el apellido materno inválido.	Nombre = "Test", ApellidoMaterno = "Test*****[--1-", ApellidoPaterno = "Test", Direccion = "Calle Test #Tes Col.Test", Email = test@gmail.com , Telefono = "2244556677", IdTurno = 1, IdPuesto = 3, Contrasena = "asdASD123*asads".	Código 400 Mensaje: Los nombres sólo pueden contener letras.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código 400 Mensaje: Los nombres sólo pueden contener letras.	Sí.	11/06/2023	

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-41	Probar el flujo alternativo para registrar a una nueva persona en el staff con el apellido paterno inválido.	Nombre = "Test", ApellidoMaterno = "Test", ApellidoPaterno = "Test*****[--1-", Direccion = "Calle Test #Tes Col.Test", Email = test@gmail.com , Telefono = "2244556677",	Código 400 Mensaje: Los nombres sólo pueden contener letras.

		IdTurno = 1, IdPuesto = 3, Contraseña = "asdASD123*asads".	
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código 400 Mensaje: Los nombres sólo pueden contener letras.	Sí.	11/06/2023	

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-42	Probar el flujo alternativo para registrar a una nueva persona en el staff con un email inválido.	Nombre = "Test", ApellidoMaterno = "Test", ApellidoPaterno = "Test", Direccion = "Calle Test #Tes Col.Test", Email = "asasdsad", Telefono = "2244556677", IdTurno = 1, IdPuesto = 3, Contraseña = "asdASD123*asads".	Código 400 Mensaje: El email no tiene el formato correcto.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código 400 Mensaje: El email no tiene el formato correcto.	Sí.	11/06/2023	

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-43	Probar el flujo alternativo para registrar a una nueva persona en el staff con un teléfono inválido.	Nombre = "Test", ApellidoMaterno = "Test", ApellidoPaterno = "Test", Direccion = "Calle Test #Tes Col.Test",	Código 400 Mensaje: El teléfono no tiene el formato correcto.

		Email = test@gmail.com , Telefono = "Test****[--1-" IdTurno = 1, IdPuesto = 3, Contrasena = "asdASD123*asads".	
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código 400 Mensaje: El teléfono no tiene el formato correcto.	Sí.	11/06/2023	

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-44	Probar el flujo alternativo para registrar a una nueva persona en el staff con una contraseña inválida.	Nombre = "Test", ApellidoMaterno = "Test", ApellidoPaterno = "Test", Direccion = "Calle Test #Tes Col.Test", Email = test@gmail.com , Telefono = "2244556677", IdTurno = 1, IdPuesto = 3, Contrasena = "hola".	Código 400 Mensaje: La contraseña debe tener al menos 8 caracteres: al menos una letra minúscula; al menos una mayúscula y al menos un número
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código 400 Mensaje: La contraseña debe tener al menos 8 caracteres: al menos una letra minúscula; al menos una mayúscula y al menos un número	Sí.	11/06/2023	

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-45	Probar el flujo alternativo para registrar a una nueva persona en el staff con un email que ya pertenece a otra persona.	Nombre = "Test", ApellidoMaterno = "Test", ApellidoPaterno = "Test", Direccion = "Calle Test #Tes Col.Test", Email = maledict@proton.me (Cualquier otro que ya este registrado en el sistema), Telefono = "2244556677", IdTurno = 1, IdPuesto = 3, Contrasena = "hola".	Código 422 Mensaje: El nombre y/o email que desea registrar ya existe en el sistema.
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código 422 Mensaje: El nombre y/o email que desea registrar ya existe en el sistema.	Sí.	11/06/2023	

Identificador Único CASO DE PRUEBA	Objetivo	Entradas	Resultados esperados
TEST-46	Probar el flujo alternativo para registrar a una nueva persona en el staff con un nombre completo que ya pertenece a otra persona.	Nombre = "Ricardo", ApellidoMaterno = "Castro", ApellidoPaterno = "Salazar", Direccion = "Calle Test #Tes Col.Test", Email = maledict@proton.me (Cualquier otro que ya este registrado en el sistema), Telefono = "2244556677", IdTurno = 1,	Código 422 Mensaje: El nombre y/o email que desea registrar ya existe en el sistema.

		IdPuesto = 3, Contrasena = "hola".	
Resultado del caso de prueba	Prueba exitosa	Fecha	
Código 422 Mensaje: El nombre y/o email que desea registrar ya existe en el sistema.	Sí.	11/06/2023	

Total de pruebas realizadas: 9	Cobertura: 100%
---------------------------------------	------------------------

5.1.9. Cobertura total de servicios

$$CoberturaTotal = \frac{Pruebas\ exitosas}{Pruebas\ Diseñadas}$$

$$CoberturaTotal = \frac{47}{47} = 100\%$$

ANEXO A. Reglas del negocio

ID	Definición	Tipo
RN-01	Cualquier persona puede consultar el menú del restaurant.	Hecho.
RN-02	Sólo el Gerente puede gestionar los productos en el menú.	Restricción
RN-03	Sólo el Gerente puede gestionar las ofertas del restaurant.	Restricción.
RN-04	Si un miembro alcanza los 20 pedidos, entonces se convierte en cliente consentido.	Habilitador de acción.
RN-05	Sólo un miembro con al menos un (1) pedido pagado puede publicar una reseña.	Restricción.
RN-06	Una imagen subida al servidor de la aplicación puede pesar hasta un (1) megabyte.	Restricción.
RN-07	Sólo el Gerente puede eliminar una reseña publicada.	Restricción.
RN-08	Sólo el Gerente puede modificar la información de un Empleado.	Restricción.
RN-09	Las ofertas se deshabilitan una vez cumplida la fecha fin de esta.	Hecho.