

Proyecto Final G92

Hito I – Diseño y prototipo

Fabián Marcelo Trarupil Fernández.

Contenido

1.- Diseño de la interfaz gráfica.....	3
A.- Página Principal.....	3
B.- Página de Formulario de Registro.....	4
C.- Inicio de Sesión	5
D.- Mi Perfil.....	6
E.- Formulario para Crear una Publicación.....	7
F.- Galería de Publicaciones (Marketplace)	8
G.- Vista de Detalle de una Publicación	9
2.- Definición de la Navegación (Públicas & Privadas)	10
3. Listado Completo de Tecnologías y Dependencias	14
4.- Diseño de las tablas de la base de datos y sus relaciones	15
5. Contrato de Datos de la API REST	26

1.- Diseño de la interfaz gráfica.

A.- Página Principal

PrematureClothes

Buscar ropa, accesorios...

Login Registro

Cuidado especial para pequeños valientes

Ropa diseñada específicamente para las necesidades de bebés prematuros.
[Ver Colección](#)

Nuestras Categorías



Recién Nacidos
Tallas especiales desde 1.5kg.



Accesorios
Gorros y calcetines sin costuras.



Cuidados Especiales
Prendas para fácil acceso a monitores.

Contacto: contacto@prematureclothes.cl
Providencia, Santiago

Instagram | Facebook

B.- Página de Formulario de Registro.

Crear Cuenta

Regístrate para cuidar lo que más amas.

Nombre Completo
Ej: Juan Pérez

Correo Electrónico
ejemplo@correo.com

Teléfono
+56921234567

Dirección
Calle/Númeración/Comuna

Contraseña
Crea una clave segura

Confirmar Contraseña
Repite tu clave

Crear Cuenta

¿Ya eres parte? [Inicia sesión](#)

C.- Inicio de Sesión

Bienvenido

Ingresa tus datos para continuar

Correo Electrónico

Contraseña

[¿Olvidaste tu contraseña?](#)

Iniciar Sesión

¿No tienes cuenta? [Regístrate aquí](#)

D.- Mi Perfil



Juan Sánchez
juan.sanchez@ejemplo.com

[Editar Datos Personales](#)

Mis Publicaciones

[Configuración](#)

[Cerrar Sesión](#)

Mis Publicaciones

2 Artículos

Venta

Body Prematuro 1.5kg
Estado: Nuevo

[Ver](#) [Eliminar](#)

Donación

Set Gorritos Algodón
Estado: Usado (Excelente)

[Ver](#) [Eliminar](#)

E.- Formulario para Crear una Publicación

Crear Publicación

Completa los datos para ayudar a otra familia.

TÍTULO DEL PRODUCTO

Ej: Body algodón orgánico

PRECIO (\$)

0 para donación

TALLA (PESO DEL BEBÉ)

Micro (Menos de 1kg)

ESTADO DE LA PRENDA

Nuevo con etiqueta

DESCRIPCIÓN

Detalla materiales, broches, o si es apto para monitores...

 SUBIR FOTOS DE LA PRENDA

Publicar Producto

F.- Galería de Publicaciones (Marketplace)

Filtrar Por

PESO / TALLA

Todas las tallas

RANGO DE PRECIO

Min Max

CATEGORÍA

Bodys
 Gorros
 Panties

Aplicar Filtros

Ropa para Prematuros

Mostrando 24 productos encontrados

Imagen

Conjunto Algodón Pima
\$15.990
Talla: Pre (1.2kg)

Pack 3 Gorritos Nudo
\$8.500
Talla: Micro (< 1kg)

Osito con Pies
\$12.000
Talla: Extra (2kg)

Imagen

Conjunto Algodón Pima
\$15.990
Talla: Pre (1.2kg)

Pack 3 Gorritos Nudo
\$8.500
Talla: Micro (< 1kg)

Osito con Pies
\$12.000
Talla: Extra (2kg)

G.- Vista de Detalle de una Publicación

Imagen en Alta Resolución

Inicio > Ropa > Bodys

Body Algodón Pima - Prematuro Extremo

\$14.990

Especificaciones de la Tela

Material: 100% Algodón Orgánico (Certificado)

Costuras: Externas para evitar roces

Broches: Libres de níquel (Hipoalergénicos)

Uso: Apto para UCIN (permite paso de cables)

Publicado por: **Antümalen G.**
★ 4.9 (15 ventas)

Añadir al Carrito

Contactar Vendedor

Envío disponible a todo Chile

2.- Definición de la Navegación (Públicas & Privadas)

Tipo	Vista	Acceso
Pública	Home / Principal	Libre
Pública	Registro de Usuarios	Libre
Pública	Inicio de Sesión	Libre
Pública	Catálogo de Productos	Libre
Pública	Detalle del Producto	Libre
Privada	Mi Perfil	Token Requerido
Privada	Editar Perfil	Token Requerido
Privada	Publicar Producto	Token Requerido
Privada	Mis Publicaciones	Token Requerido
Privada	Carrito de Compras	Token Requerido
Privada	Mis Pedidos	Token Requerido
Privada	Favoritos	Token Requerido
Privada	Mensajes	Token Requerido
Privada	Reportes de Ventas	Token Requerido
Privada	Gestión Usuarios	Token Requerido
Privada	Gestión Pedidos	Token Requerido
Privada	Estadísticas Vendedor	Token Requerido

config:

layout: elk

flowchart TB

subgraph subGraph0["ACCESO: LIBRE (Sin restricciones)"]

direction TB

Home("Home / Página Principal")

Registro("Registro de Usuarios")

Login("Inicio de Sesión")

Catalogo("Catálogo de Productos")

Detalle("Detalle del Producto")

```
Publica["Sección Pública"]  
end  
  
subgraph subGraph1["ACCESO: TOKEN REQUERIDO (Logueado)"]  
    direction TB  
  
    Perfil("Mi Perfil")  
    EditarPerfil("Editar Perfil")  
    Publicar("Publicar Producto")  
    MisPublic("Mis Publicaciones")  
    Carrito("Carrito de Compras")  
    Pedidos("Mis Pedidos")  
    Favoritos("Favoritos")  
    Mensajes("Mensajes")  
    Privada["Sección Privada"]  
end  
  
subgraph subGraph2["ACCESO: ADMINISTRADOR"]  
    direction TB  
  
    AdminUsu("Gestión Usuarios")  
    AdminPed("Gestión Pedidos")  
    Reportes("Reportes de Ventas")  
    Admin["Sección Administración"]  
end  
  
subgraph subGraph3["ACCESO: VENDEDOR"]  
    direction TB  
  
    Estadisticas("Estadísticas Vendedor")  
    Vendedor["Sección Vendedor"]  
end
```

Root["Marketplace Ropa Bebés Prematuros"] -- Se divide en --> Publica & Privada

Publica --- Home & Registro & Login & Catalogo

Catalogo .. Clic en producto .-> Detalle

Privada --- Perfil & EditarPerfil & Publicar & MisPublic & Carrito & Pedidos & Favoritos & Mensajes

Privada -- Roles especiales --> Admin & Vendedor

Admin --- AdminUsu & AdminPed & Reportes

Vendedor --- Estadisticas

Root:::mainNode

Publica:::typeNode

Privada:::typeNode

Admin:::roleNode

Vendedor:::roleNode

Home:::publicNode

Registro:::publicNode

Login:::publicNode

Catalogo:::publicNode

Detalle:::publicNode

Perfil:::privateNode

EditarPerfil:::privateNode

Publicar:::privateNode

MisPublic:::privateNode

Carrito:::privateNode

Pedidos:::privateNode

Favoritos:::privateNode

Mensajes:::privateNode

AdminUsu:::adminNode

AdminPed:::adminNode

Reportes:::adminNode

Estadisticas:::vendedorNode

classDef mainNode fill:#ffeb3b,stroke:#fbc02d,stroke-width:2px,color:#000,font-weight:bold

classDef typeNode fill:#e3f2fd,stroke:#2196f3,stroke-width:2px,color:#000

classDef roleNode fill:#f1f8e9,stroke:#7cb342,stroke-width:2px,color:#000

classDef publicNode fill:#fff,stroke:#4caf50,stroke-width:1px

classDef privateNode fill:#fff,stroke:#ff9800,stroke-width:1px

classDef adminNode fill:#fff,stroke:#f44336,stroke-width:1px

classDef vendedorNode fill:#fff,stroke:#9c27b0,stroke-width:1px

3. Listado Completo de Tecnologías y Dependencias

A. Desarrollo Frontend (Interfaz de Usuario)

- Lenguajes: HTML5, CSS3, JavaScript (ES6+).
- Framework Principal: React.js (Biblioteca para interfaces basadas en componentes).
- Herramienta de Construcción: Vite (Build tool de alto rendimiento).
- Dependencias de Navegación y Estilo:
 - React Router DOM: Gestión de rutas y navegación SPA.
 - Bootstrap 5 / React-Bootstrap: Framework de CSS y componentes UI responsivos.
 - Lucide-React / FontAwesome: Librerías de iconos.
- Dependencias de Comunicación:
 - Axios: Cliente HTTP para conectar con el Backend.

B. Desarrollo Backend

- Entorno de Ejecución: Node.js (Permite ejecutar JavaScript en el servidor).
- Framework de Servidor: Express.js (Para la creación de la API REST).
- Dependencias de Gestión de Datos:
 - CORS: Para permitir la comunicación segura entre el Frontend y el Backend.
 - Dotenv: Gestión de variables de entorno (seguridad de credenciales).
 - Nodemon: (Dependencia de desarrollo) Para reinicio automático del servidor al detectar cambios.
- Base de Datos :
 - PostgreSQL.

4.- Diseño de las tablas de la base de datos y sus relaciones

```
// =====  
// PROYECTO: PrematureClothes  
// VERSION: 2.0 - Profesional Extendida  
// =====
```

```
Project PrematureClothes {  
    database_type: 'PostgreSQL'  
  
    Note: ""  
        Marketplace especializado en ropa para bebés prematuros.  
        Esquema optimizado para producción con funcionalidades completas de marketplace.  
        ""  
}  
}
```

```
// --- ENUMS AMPLIADOS ---
```

```
Enum rol_usuario {  
    comprador  
    vendedor  
    admin  
}
```

```
Enum talla_prematuro {  
    "000"  
    "00"  
    "0"  
    RN
```

```
}
```

```
Enum estado_prenda {
```

```
    nuevo
```

```
    usado_como_nuevo
```

```
    usado
```

```
}
```

```
Enum metodo_pago {
```

```
    Webpay
```

```
    Paypal
```

```
    Transferencia
```

```
}
```

```
Enum estado_pedido {
```

```
    pendiente
```

```
    pagado
```

```
    enviado
```

```
    entregado
```

```
    cancelado
```

```
}
```

```
Enum tipo_notificacion {
```

```
    nuevo_mensaje
```

```
    pedido_actualizado
```

```
    producto_favorito
```

```
nuevo_comentario  
sistema  
}
```

```
// --- TABLAS MODIFICADAS Y NUEVAS ---
```

```
Table usuarios {  
    id integer [primary key, increment]  
    nombre_completo varchar(100) [not null]  
    email varchar(100) [unique, not null, note: 'Validar formato email']  
    password varchar(255) [not null, note: 'Hash bcrypt/scrypt']  
    direccion_envio text  
    latitud decimal(10, 8) [note: 'Geolocalización para búsquedas']  
    longitud decimal(11, 8) [note: 'Geolocalización para búsquedas']  
    rol rol_usuario [default: 'comprador']  
    activo boolean [default: true]  
    created_at timestamp [default: `now()`]  
    updated_at timestamp [default: `now()`]
```

```
indexes {  
    email [unique]  
    activo  
    rol  
    (latitud, longitud) [name: 'idx_usuarios ubicacion']  
}
```

```
Table categorias {  
    id integer [primary key, increment]  
    nombre varchar(50) [unique, not null]  
    descripcion text  
    created_at timestamp [default: `now()`]  
    updated_at timestamp [default: `now()`]  
}  
  
}
```

```
Table productos {  
    id integer [primary key, increment]  
    vendedor_id integer [not null]  
    categoria_id integer [not null]  
    titulo varchar(100) [not null]  
    descripcion text  
    precio integer [not null, note: 'CHECK precio > 0']  
    stock integer [not null, default: 0, note: 'CHECK stock >= 0']  
    talla_rango talla_prematuro [not null]  
    estado_prenda estado_prenda [not null]  
    activo boolean [default: true, note: 'Baja lógica: DELETE marca activo = false']  
    created_at timestamp [default: `now()`]  
    updated_at timestamp [default: `now()`]  
}
```

```
indexes {  
    vendedor_id  
    categoria_id
```

```
    precio  
    talla_rango  
    (categoria_id, activo) [name: 'idx_productos_cat_act']  
    created_at [name: 'idx_productos_recientes', note: 'Orden descendente']  
}  
}
```

```
Table imagenes_producto {  
    id integer [primary key, increment]  
    producto_id integer [not null]  
    url text [not null, note: 'S3/Cloudinary URL']  
    orden integer [not null, default: 1, note: 'Orden de visualización']  
    created_at timestamp [default: `now()`]}
```

```
indexes {  
    producto_id  
    orden  
}  
}
```

```
Table historial_precios {  
    id integer [primary key, increment]  
    producto_id integer [not null]  
    precio_anterior integer [not null]  
    precio_nuevo integer [not null]  
    fecha_cambio timestamp [default: `now()`]}
```

```
indexes {  
    producto_id  
    fecha_cambio  
}  
}
```

```
Table carrito_items {  
  
    id integer [primary key, increment]  
  
    usuario_id integer [not null]  
  
    producto_id integer [not null]  
  
    cantidad integer [not null, default: 1, note: 'CHECK cantidad >= 1']  
  
    created_at timestamp [default: `now()`]  
  
    updated_at timestamp [default: `now()`]
```

```
indexes {  
    (usuario_id, producto_id) [unique]  
  
    usuario_id  
}  
}
```

```
Table pedidos {  
  
    id integer [primary key, increment]  
  
    usuario_id integer [not null]  
  
    total integer [not null, default: 0, note: 'Cálculo en App: SUM(detalles)']  
  
    estado estado_pedido [default: 'pendiente']
```

```
    metodo_pago metodo_pago [not null]
    direccion_entrega text [not null, note: 'Validar minLength 10']
    fecha_pedido timestamp [default: `now()`]
    updated_at timestamp [default: `now()`]
```

```
indexes {
    usuario_id
    estado
    (usuario_id, estado)
    fecha_pedido [name: 'idx_pedidos_recientes']
}
```

}

```
Table pedido_detalles {
    id integer [primary key, increment]
    pedido_id integer [not null]
    producto_id integer [not null]
    producto_titulo varchar(100) [not null, note: 'Snapshot inmutable']
    cantidad integer [not null, note: 'CHECK > 0']
    precio_unitario integer [not null, note: 'CHECK > 0']
    created_at timestamp [default: `now()`]
```

Note: 'Snapshot inmutable de productos al momento de la compra.'

}

```
Table favoritos {
```

```
id integer [primary key, increment]
usuario_id integer [not null]
producto_id integer [not null]
created_at timestamp [default: `now()`]
```

```
indexes {
  (usuario_id, producto_id) [unique]
}
}
```

```
Table comentarios {
  id integer [primary key, increment]
  usuario_id integer [not null]
  producto_id integer [not null]
  contenido text [not null, note: 'Validar minLength 5']
  calificacion integer [not null, note: 'CHECK 1-5']
  created_at timestamp [default: `now()`]
  updated_at timestamp [default: `now()`]
```

```
indexes {
  producto_id
  calificacion
  created_at [name: 'idx_comentarios_recientes']
}
}
```

```
Table mensajes {  
    id integer [primary key, increment]  
    pedido_id integer [note: 'Contexto de la conversación']  
    remitente_id integer [not null]  
    destinatario_id integer [not null]  
    contenido text [not null]  
    leido boolean [default: false]  
    created_at timestamp [default: `now()`]  
  
    indexes {  
        pedido_id  
        remitente_id  
        destinatario_id  
        (remitente_id, destinatario_id, created_at) [name: 'idx_mensajes_conversacion']  
    }  
}
```

```
Table notificaciones {  
    id integer [primary key, increment]  
    usuario_id integer [not null]  
    tipo tipo_notificacion [not null]  
    contenido text [not null]  
    metadata jsonb [note: 'Datos adicionales (ej: {producto_id: 123})']  
    leida boolean [default: false]  
    created_at timestamp [default: `now()`]
```

```
indexes {
    usuario_id
    leida
    (usuario_id, leida)
    created_at [name: 'idx_notificaciones_recientes']
}

}

// --- RELACIONES CON POLÍTICAS ON DELETE OPTIMIZADAS ---
Ref: productos.vendedor_id > usuarios.id [delete: restrict]
Ref: productos.categoría_id > categorías.id [delete: restrict]
Ref: imágenes_producto.producto_id > productos.id [delete: cascade]
Ref: historial_precios.producto_id > productos.id [delete: cascade]
Ref: carrito_items.usuario_id > usuarios.id [delete: cascade]
Ref: carrito_items.producto_id > productos.id [delete: cascade]
Ref: pedidos.usuario_id > usuarios.id [delete: restrict]
Ref: pedido_detalles.pedido_id > pedidos.id [delete: cascade]
Ref: pedido_detalles.producto_id > productos.id [delete: restrict]
Ref: favoritos.usuario_id > usuarios.id [delete: cascade]
Ref: favoritos.producto_id > productos.id [delete: cascade]
Ref: comentarios.usuario_id > usuarios.id [delete: cascade]
Ref: comentarios.producto_id > productos.id [delete: cascade]
Ref: mensajes.pedido_id > pedidos.id [delete: cascade]
Ref: mensajes.remitente_id > usuarios.id [delete: cascade]
Ref: mensajes.destinatario_id > usuarios.id [delete: cascade]
Ref: notificaciones.usuario_id > usuarios.id [delete: cascade]
```


5. Contrato de Datos de la API REST

Documentación técnica sistema de gestión de productos y usuarios para}

openapi: 3.0.0

info:

title: PrematureClothes Marketplace API

version: 2.3.1

description: |

API REST profesional para marketplace especializado en ropa para bebés prematuros.

Incluye funcionalidades completas de e-commerce, mensajería y geolocalización.

contact:

name: Equipo PrematureClothes

email: soporte@prematureclothes.com

license:

name: MIT

url: <https://opensource.org/licenses/MIT>

servers:

Added by API Auto Mocking Plugin

- description: SwaggerHub API Auto Mocking

url: <https://virtserver.swaggerhub.com/ftwebdeveloper/MarketplaceRopaBebesPrematuros/2.3.1>

- url: <https://api.prematureclothes.com/v1>

description: Servidor de producción

- url: <https://staging-api.prematureclothes.com/v1>

description: Servidor de staging

- url: <http://localhost:3000/v1>

description: Servidor local de desarrollo

security:

- bearerAuth: []

tags:

- name: Autenticación

description: Registro, login y gestión de sesión

- name: Usuarios

description: Gestión de perfiles y ubicación

- name: Marketplace

description: Catálogo y gestión de productos

- name: Carrito

description: Gestión del carrito de compras

- name: Pedidos

description: Checkout y gestión de pedidos

- name: Interacción

description: Favoritos, comentarios y mensajería

- name: Administración

description: Endpoints exclusivos para administradores

paths:

/auth/registro:

post:

tags:

- Autenticación

summary: Registrar nuevo usuario

description: Crea una nueva cuenta en la plataforma

security: []

requestBody:

required: true

content:

application/json:

schema:

\$ref: '#/components/schemas/UsuarioRegistro'

example:

nombre_completo: "Ana García López"

email: "ana@ejemplo.com"

password: "SecurePass123"

direccion_envio: "Calle Principal 123, Santiago"

latitud: -33.4489

longitud: -70.6693

responses:

'201':

description: Usuario registrado exitosamente

content:

application/json:

schema:

\$ref: '#/components/schemas/UsuarioPerfil'

'400':

\$ref: '#/components/responses/BadRequest'

'409':

\$ref: '#/components/responses/Conflict'

```
/auth/login:  
  post:  
    tags:  
      - Autenticación  
    summary: Iniciar sesión  
    description: Autentica usuario y devuelve token JWT  
    security: []  
    requestBody:  
      required: true  
      content:  
        application/json:  
          schema:  
            type: object  
            required:  
              - email  
              - password  
            properties:  
              email:  
                type: string  
                format: email  
                example: "ana@ejemplo.com"  
              password:  
                type: string  
                format: password  
                example: "SecurePass123"
```

responses:

'200':

 description: Login exitoso

 content:

 application/json:

 schema:

 type: object

 properties:

 token:

 type: string

 example: "eyJhbGciOiJIUzI1NilsInR5cCI6IkpXVCJ9..."

 usuario:

 \$ref: '#/components/schemas/UsuarioPerfil'

'401':

 \$ref: '#/components/responses/Unauthorized'

/usuarios/me:

get:

 tags:

 - Usuarios

 summary: Obtener mi perfil

responses:

'200':

 content:

 application/json:

 schema:

```
$ref: '#/components/schemas/UsuarioPerfilCompleto'  
'401':  
  $ref: '#/components/responses/Unauthorized'  
  
put:  
  tags:  
    - Usuarios  
  summary: Actualizar mi perfil  
  requestBody:  
    content:  
      application/json:  
        schema:  
          $ref: '#/components/schemas/UsuarioUpdate'  
        example:  
          nombre_completo: "Ana García López Actualizado"  
          direccion_envio: "Calle Nueva 456, Santiago"  
  responses:  
    '200':  
      content:  
        application/json:  
          schema:  
            $ref: '#/components/schemas/UsuarioPerfilCompleto'  
    '400':  
      $ref: '#/components/responses/BadRequest'  
  
  
/usuarios/ubicacion:  
patch:
```

tags:

- Usuarios

summary: Actualizar ubicación

description: Actualiza las coordenadas de geolocalización del usuario

requestBody:

required: true

content:

application/json:

schema:

type: object

required:

- latitud

- longitud

properties:

latitud:

type: number

format: float

example: -33.4489

longitud:

type: number

format: float

example: -70.6693

responses:

'200':

description: Ubicación actualizada

'400':

\$ref: '#/components/responses/BadRequest'

/productos:

get:

tags:

- Marketplace

summary: Buscar productos con filtros avanzados

description: |

Catálogo paginado con múltiples filtros y búsqueda por proximidad.

Incluye imágenes múltiples por producto.

security: []

parameters:

- \$ref: '#/components/parameters/PageParam'

- \$ref: '#/components/parameters/LimitParam'

- name: categoria_id

- in: query

- schema:

- type: integer

- description: Filtrar por categoría

- name: precio_min

- in: query

- schema:

- type: integer

- minimum: 0

- description: Precio mínimo

- name: precio_max

```
in: query
schema:
  type: integer
description: Precio máximo
- name: talla_rango
  in: query
  schema:
    type: string
  enum:
    - "000"
    - "00"
    - "0"
    - "RN"
description: Filtrar por talla
- name: estado_prenda
  in: query
  schema:
    type: string
  enum:
    - nuevo
    - usado_como_nuevo
    - usado
description: Estado de la prenda
- name: latitud
  in: query
  schema:
```

type: number
format: float
description: Tu latitud para búsqueda por proximidad

- name: longitud
in: query
schema:
type: number
format: float
description: Tu longitud para búsqueda por proximidad

- name: radio_km
in: query
schema:
type: integer
minimum: 1
default: 10
description: Radio de búsqueda en kilómetros

- name: orden
in: query
schema:
type: string
enum:

- precio_asc
- precio_desc
- reciente
- cercania

default: reciente

description: Orden de resultados

responses:

'200':

description: Lista paginada de productos

headers:

X-Total-Count:

schema:

type: integer

description: Total de productos disponibles con los filtros aplicados

content:

application/json:

schema:

type: object

properties:

total:

type: integer

pagina:

type: integer

total_paginas:

type: integer

data:

type: array

items:

\$ref: '#/components/schemas/ProductoDetalleCompleto'

post:

tags:

- Marketplace

summary: Publicar nuevo producto

requestBody:

required: true

content:

application/json:

schema:

\$ref: '#/components/schemas/ProductoInput'

example:

titulo: "Body prematuro algodón orgánico"

descripcion: "Body ideal para bebés prematuros, talla 000, algodón 100% orgánico"

precio: 12990

stock: 5

categoria_id: 1

talla_rango: "000"

estado_prenda: "nuevo"

imagenes:

- "https://cdn.com/body1.jpg"

- "https://cdn.com/body2.jpg"

responses:

'201':

description: Producto publicado exitosamente

content:

application/json:

schema:

\$ref: '#/components/schemas/ProductoDetalleCompleto'

```
'400':  
    $ref: '#/components/responses/BadRequest'  
  
'403':  
    $ref: '#/components/responses/Forbidden'  
  
  
/productos/{id}:  
get:  
tags:  
- Marketplace  
summary: Obtener detalle de producto  
parameters:  
- $ref: '#/components/parameters/IdParam'  
security: []  
responses:  
'200':  
content:  
application/json:  
schema:  
$ref: '#/components/schemas/ProductoDetalleCompleto'  
'404':  
$ref: '#/components/responses/NotFound'  
put:  
tags:  
- Marketplace  
summary: Editar producto  
parameters:
```

- \$ref: '#/components/parameters/IdParam'

requestBody:

content:

application/json:

schema:

\$ref: '#/components/schemas/ProductoUpdate'

responses:

'200':

content:

application/json:

schema:

\$ref: '#/components/schemas/ProductoDetalleCompleto'

'403':

\$ref: '#/components/responses/Forbidden'

'404':

\$ref: '#/components/responses/NotFound'

delete:

tags:

- Marketplace

summary: Eliminar producto (baja lógica)

parameters:

- \$ref: '#/components/parameters/IdParam'

responses:

'204':

description: Producto marcado como inactivo

'403':

```
$ref: '#/components/responses/Forbidden'  
'404':  
$ref: '#/components/responses/NotFound'
```

/productos/{id}/imagenes:

post:

tags:

- Marketplace

summary: Añadir imágenes a producto

parameters:

- \$ref: '#/components/parameters/IdParam'

requestBody:

required: true

content:

application/json:

schema:

type: object

required:

- imagenes

properties:

imagenes:

type: array

items:

type: string

format: uri

minItems: 1

responses:

'201':

description: Imágenes añadidas exitosamente

'403':

\$ref: '#/components/responses/Forbidden'

get:

tags:

- Marketplace

summary: Obtener imágenes del producto

parameters:

- \$ref: '#/components/parameters/IdParam'

security: []

responses:

'200':

content:

application/json:

schema:

type: array

items:

\$ref: '#/components/schemas/ImagenProducto'

/productos/{id}/imagenes/{imagen_id}:

delete:

tags:

- Marketplace

summary: Eliminar imagen de producto

parameters:

- \$ref: '#/components/parameters/IdParam'

- name: imagen_id

in: path

required: true

schema:

type: integer

responses:

'204':

description: Imagen eliminada

'403':

\$ref: '#/components/responses/Forbidden'

'404':

\$ref: '#/components/responses/NotFound'

/productos/mis-publicaciones:

get:

tags:

- Marketplace

summary: Mis productos publicados

responses:

'200':

content:

application/json:

schema:

type: array

```
items:  
  $ref: '#/components/schemas/ProductoDetalleCompleto'  
  
/carrito:  
  get:  
    tags:  
      - Carrito  
    summary: Ver mi carrito  
    responses:  
      '200':  
        content:  
          application/json:  
            schema:  
              type: array  
              items:  
                $ref: '#/components/schemas/ItemCarritoCompleto'  
  post:  
    tags:  
      - Carrito  
    summary: Añadir producto al carrito  
    requestBody:  
      required: true  
      content:  
        application/json:  
          schema:  
            type: object
```

required:

- producto_id

- cantidad

properties:

producto_id:

 type: integer

cantidad:

 type: integer

 minimum: 1

responses:

'201':

content:

 application/json:

 schema:

 \$ref: '#/components/schemas/ItemCarritoCompleto'

/carrito/{id}:

delete:

tags:

- Carrito

summary: Eliminar item del carrito

parameters:

- \$ref: '#/components/parameters/IdParam'

responses:

'204':

description: Item eliminado del carrito

```
patch:  
tags:  
  - Carrito  
summary: Actualizar cantidad en carrito  
parameters:  
  - $ref: '#/components/parameters/IdParam'  
requestBody:  
  required: true  
  content:  
    application/json:  
      schema:  
        type: object  
        required:  
          - cantidad  
        properties:  
          cantidad:  
            type: integer  
            minimum: 1  
responses:  
  '200':  
    content:  
      application/json:  
        schema:  
          $ref: '#/components/schemas/ItemCarritoCompleto'  
  
/pedidos:
```

get:

tags:

- Pedidos

summary: Mis pedidos

responses:

'200':

content:

application/json:

schema:

type: array

items:

\$ref: '#/components/schemas/PedidoCompleto'

post:

tags:

- Pedidos

summary: Realizar checkout

description: Convierte el carrito actual en un pedido

requestBody:

required: true

content:

application/json:

schema:

\$ref: '#/components/schemas/PedidoInput'

example:

metodo_pago: "Webpay"

direccion_entrega: "Calle Principal 123, Santiago, Chile"

responses:

'201':

content:

application/json:

schema:

\$ref: '#/components/schemas/PedidoCompleto'

'400':

\$ref: '#/components/responses/BadRequest'

'409':

description: Stock insuficiente para algunos productos

/pedidos/{id}:

get:

tags:

- Pedidos

summary: Detalle de pedido

parameters:

- \$ref: '#/components/parameters/IdParam'

responses:

'200':

content:

application/json:

schema:

\$ref: '#/components/schemas/PedidoCompleto'

'403':

\$ref: '#/components/responses/Forbidden'

/pedidos/{id}/estado:

patch:

tags:

- Pedidos

summary: Actualizar estado del pedido (Vendedor/Admin)

parameters:

- \$ref: '#/components/parameters/IdParam'

requestBody:

required: true

content:

application/json:

schema:

type: object

required:

- estado

properties:

estado:

type: string

enum:

- pagado
- enviado
- entregado
- cancelado

responses:

'200':

```
description: Estado actualizado  
'403':  
  $ref: '#/components/responses/Forbidden'  
  
  
/favoritos:  
  get:  
    tags:  
      - Interacción  
    summary: Mis productos favoritos  
    responses:  
      '200':  
        content:  
          application/json:  
            schema:  
              type: array  
              items:  
                $ref: '#/components/schemas/ProductoDetalleCompleto'  
      post:  
        tags:  
          - Interacción  
        summary: Añadir a favoritos  
        requestBody:  
          required: true  
          content:  
            application/json:  
              schema:
```

```
type: object
required:
- producto_id
properties:
producto_id:
  type: integer
responses:
'201':
  description: Añadido a favoritos
  content:
    application/json:
      schema:
        type: object
        properties:
id:
  type: integer
producto:
  $ref: '#/components/schemas/ProductoDetalleCompleto'

/favoritos/{id}:
delete:
tags:
- Interacción
summary: Eliminar de favoritos
parameters:
- $ref: '#/components/parameters/IdParam'
```

responses:

'204':

description: Eliminado de favoritos

/comentarios:

post:

tags:

- Interacción

summary: Crear comentario/reseña

requestBody:

required: true

content:

application/json:

schema:

\$ref: '#/components/schemas/ComentarioInput'

example:

producto_id: 1

contenido: "Excelente calidad, perfecto para mi bebé prematuro"

calificacion: 5

responses:

'201':

description: Comentario creado

content:

application/json:

schema:

\$ref: '#/components/schemas/ComentarioOutput'

/comentarios/{id}:

put:

tags:

- Interacción

summary: Editar comentario

parameters:

- \$ref: '#/components/parameters/IdParam'

requestBody:

content:

application/json:

schema:

\$ref: '#/components/schemas/ComentarioUpdate'

responses:

'200':

content:

application/json:

schema:

\$ref: '#/components/schemas/ComentarioOutput'

'403':

\$ref: '#/components/responses/Forbidden'

delete:

tags:

- Interacción

summary: Eliminar comentario

parameters:

- \$ref: '#/components/parameters/IdParam'

responses:

'204':

description: Comentario eliminado

'403':

\$ref: '#/components/responses/Forbidden'

/productos/{id}/comentarios:

get:

tags:

- Interacción

summary: Comentarios de un producto

parameters:

- \$ref: '#/components/parameters/IdParam'

- \$ref: '#/components/parameters/PageParam'

- \$ref: '#/components/parameters/LimitParam'

security: []

responses:

'200':

content:

application/json:

schema:

type: object

properties:

promedio_calificacion:

type: number

```
    format: float

total_comentarios:
    type: integer

comentarios:
    type: array
    items:
        $ref: '#/components/schemas/ComentarioOutput'
```

```
/mensajes:
get:
tags:
- Interacción
summary: Obtener conversaciones
description: Lista las conversaciones activas del usuario
```

```
responses:
  '200':
content:
  application/json:
    schema:
      type: array
      items:
        $ref: '#/components/schemas/Conversacion'
```

```
post:
tags:
- Interacción
summary: Enviar mensaje
```

```
requestBody:  
  required: true  
  
  content:  
    application/json:  
      schema:  
        $ref: '#/components/schemas/MensajeInput'  
  
      example:  
        destinatario_id: 2  
        pedido_id: 1  
        contenido: "Hola, ¿este body viene en talla 00?"  
  
responses:  
  '201':  
    content:  
      application/json:  
        schema:  
          $ref: '#/components/schemas/MensajeOutput'
```

/mensajes/conversacion/{usuario_id}:

get:

tags:

- Interacción

summary: Obtener historial de conversación

parameters:

- name: usuario_id
 in: path
 required: true

```
schema:  
  type: integer  
  - $ref: '#/components/parameters/PageParam'  
  - $ref: '#/components/parameters/LimitParam'
```

responses:

'200':

content:

application/json:

schema:

type: array

items:

\$ref: '#/components/schemas/MensajeOutput'

/mensajes/{id}/leido:

put:

tags:

- Interacción

summary: Marcar mensaje como leído

parameters:

- \$ref: '#/components/parameters/IdParam'

responses:

'200':

description: Mensaje marcado como leído

/notificaciones:

get:

tags:

- Interacción

summary: Mis notificaciones

parameters:

- name: leidas

 in: query

 schema:

 type: boolean

 description: Filtrar por estado de lectura

responses:

'200':

 content:

 application/json:

 schema:

 type: array

 items:

 \$ref: '#/components/schemas/Notificacion'

/notificaciones/{id}/leida:

put:

tags:

- Interacción

summary: Marcar notificación como leída

parameters:

- \$ref: '#/components/parameters/IdParam'

responses:

'200':

description: Notificación marcada como leída

/admin/usuarios:

get:

tags:

- Administración

summary: (ADMIN) Listar todos los usuarios

parameters:

- \$ref: '#/components/parameters/PageParam'
- \$ref: '#/components/parameters/LimitParam'
- name: rol

in: query

schema:

type: string

enum:

- comprador
- vendedor
- admin

- name: activo

in: query

schema:

type: boolean

responses:

'200':

content:

```
application/json:  
  schema:  
    type: object  
    properties:  
      total:  
        type: integer  
      data:  
        type: array  
        items:  
          $ref: '#/components/schemas/UsuarioPerfilCompleto'
```

```
/admin/pedidos:  
  get:  
    tags:  
      - Administración  
    summary: (ADMIN) Listar todos los pedidos  
    parameters:  
      - $ref: '#/components/parameters/PageParam'  
      - $ref: '#/components/parameters/LimitParam'  
      - name: estado  
        in: query  
    schema:  
      type: string  
      enum:  
        - pendiente  
        - pagado
```

- enviado
- entregado
- cancelado

- name: fecha_desde

in: query

schema:

 type: string

 format: date

- name: fecha_hasta

in: query

schema:

 type: string

 format: date

responses:

'200':

 content:

 application/json:

 schema:

 type: object

 properties:

 total:

 type: integer

 data:

 type: array

 items:

 \$ref: '#/components/schemas/PedidoCompleto'

/admin/reportes/ventas:

get:

tags:

- Administración

summary: (ADMIN) Reporte de ventas

parameters:

- name: fecha_desde

in: query

required: true

schema:

type: string

format: date

example: "2024-01-01"

- name: fecha_hasta

in: query

required: true

schema:

type: string

format: date

example: "2024-12-31"

responses:

'200':

content:

application/json:

schema:

\$ref: '#/components/schemas/ReporteVentas'

/vendedor/estadisticas:

get:

tags:

- Administración

summary: (VENDEDOR) Mis estadísticas

responses:

'200':

content:

application/json:

schema:

\$ref: '#/components/schemas/EstadisticasVendedor'

components:

securitySchemes:

bearerAuth:

type: http

scheme: bearer

bearerFormat: JWT

description: |

Incluir el token JWT obtenido en /auth/login.

Formato: `Authorization: Bearer <token>`

parameters:

IdParam:

```
name: id  
in: path  
required: true  
schema:  
  type: integer  
  minimum: 1  
description: ID del recurso
```

```
PageParam:  
  name: page  
  in: query  
  schema:  
    type: integer  
    minimum: 1  
    default: 1  
  description: Número de página para paginación
```

```
LimitParam:  
  name: limit  
  in: query  
  schema:  
    type: integer  
    minimum: 1  
    maximum: 100  
    default: 10  
  description: Cantidad de resultados por página
```

responses:

BadRequest:

description: Solicitud incorrecta o datos inválidos

content:

application/json:

schema:

\$ref: '#/components/schemas/ErrorResponse'

Unauthorized:

description: No autorizado. Token inválido o ausente

content:

application/json:

schema:

\$ref: '#/components/schemas/ErrorResponse'

Forbidden:

description: No tiene permisos para realizar esta acción

content:

application/json:

schema:

\$ref: '#/components/schemas/ErrorResponse'

NotFound:

description: El recurso solicitado no existe

content:

application/json:

schema:

\$ref: '#/components/schemas/ErrorResponse'

schemas:

ErrorResponse:

 type: object

 properties:

 codigo:

 type: integer

 example: 400

 mensaje:

 type: string

 example: "Datos de entrada inválidos"

 detalles:

 type: array

 items:

 type: object

 properties:

 campo:

 type: string

 error:

 type: string

 timestamp:

 type: string

 format: date-time

UsuarioRegistro:

 type: object

required:

- nombre_completo
- email
- password

properties:

nombre_completo:

 type: string
 minLength: 3
 maxLength: 100
 example: "Ana García López"

email:

 type: string
 format: email
 maxLength: 100
 example: "ana@ejemplo.com"

password:

 type: string
 minLength: 8
 format: password
 example: "SecurePass123"

direccion_envio:

 type: string
 example: "Calle Principal 123, Santiago"

latitud:

 type: number
 format: float

example: -33.4489

longitud:

type: number

format: float

example: -70.6693

rol:

type: string

enum:

- comprador

- vendedor

- admin

default: comprador

UsuarioUpdate:

type: object

properties:

nombre_completo:

type: string

minLength: 3

maxLength: 100

direccion_envio:

type: string

password:

type: string

minLength: 8

format: password

UsuarioPerfil:

type: object

properties:

id:

type: integer

example: 1

nombre_completo:

type: string

example: "Ana García López"

email:

type: string

example: "ana@ejemplo.com"

direccion_envio:

type: string

example: "Calle Principal 123, Santiago"

rol:

type: string

example: "comprador"

UsuarioPerfilCompleto:

allOf:

- \$ref: '#/components/schemas/UsuarioPerfil'

- type: object

properties:

latitud:

type: number

format: float

longitud:

type: number

format: float

activo:

type: boolean

created_at:

type: string

format: date-time

updated_at:

type: string

format: date-time

ProductоТInput:

type: object

required:

- titulo
- precio
- stock
- categoria_id
- talla_rango
- estado_prenda

properties:

titulo:

type: string

minLength: 3

maxLength: 100

example: "Body prematuro algodón orgánico"

descripcion:

type: string

example: "Body ideal para bebés prematuros..."

precio:

type: integer

minimum: 1

example: 12990

stock:

type: integer

minimum: 0

example: 5

categoria_id:

type: integer

example: 1

talla_rango:

type: string

enum:

- "000"

- "00"

- "0"

- "RN"

example: "000"

estado_prenda:

type: string

enum:

- nuevo

- usado_como_nuevo

- usado

example: "nuevo"

imagenes:

type: array

items:

type: string

format: uri

minItems: 1

example:

- "https://cdn.com/body1.jpg"

- "https://cdn.com/body2.jpg"

ProductoUpdate:

type: object

properties:

titulo:

type: string

minLength: 3

maxLength: 100

descripcion:

type: string

precio:

type: integer

minimum: 1

stock:

type: integer

minimum: 0

categoria_id:

type: integer

talla_rango:

type: string

enum:

- "000"

- "00"

- "0"

- "RN"

estado_prenda:

type: string

enum:

- nuevo

- usado_como_nuevo

- usado

activo:

type: boolean

ProductoDetalle:

type: object

properties:

id:

type: integer

example: 1

vendedor_id:

type: integer

example: 2

titulo:

type: string

example: "Body prematuro algodón orgánico"

descripcion:

type: string

example: "Body ideal para bebés prematuros..."

precio:

type: integer

example: 12990

stock:

type: integer

example: 5

talla_rango:

type: string

example: "000"

estado_prenda:

type: string

example: "nuevo"

categoria:

type: object

properties:

id:

type: integer

nombre:

type: string

ProductoDetalleCompleto:

allOf:

- \$ref: '#/components/schemas/ProductoDetalle'

- type: object

properties:

vendedor:

\$ref: '#/components/schemas/UsuarioPerfil'

imagenes:

type: array

items:

\$ref: '#/components/schemas/ImagenProducto'

promedio_calificacion:

type: number

format: float

total_comentarios:

type: integer

distancia_km:

type: number

format: float

description: Distancia al vendedor (solo en búsqueda por proximidad)

```
created_at:  
  type: string  
  format: date-time  
  
updated_at:  
  type: string  
  format: date-time
```

ImagenProducto:

```
type: object  
  
properties:  
  
  id:  
    type: integer  
  
  url:  
    type: string  
    format: uri  
  
  orden:  
    type: integer  
  
  created_at:  
    type: string  
    format: date-time
```

ItemCarrito:

```
type: object  
  
properties:  
  
  id:  
    type: integer
```

```
producto_id:  
    type: integer  
  
cantidad:  
    type: integer  
  
created_at:  
    type: string  
    format: date-time
```

```
ItemCarritoCompleto:  
allOf:  
    - $ref: '#/components/schemas/ItemCarrito'  
    - type: object  
  
properties:  
    producto:  
        $ref: '#/components/schemas/ProductoDetalleCompleto'  
  
    subtotal:  
        type: integer  
  
    description: precio * cantidad
```

```
PedidoInput:  
type: object  
  
required:  
    - metodo_pago  
    - direccion_entrega  
  
properties:  
    metodo_pago:
```

type: string

enum:

- Webpay

- Paypal

- Transferencia

example: "Webpay"

direccion_entrega:

type: string

minLength: 10

example: "Calle Principal 123, Santiago, Chile"

PedidoOutput:

type: object

properties:

id:

type: integer

usuario_id:

type: integer

total:

type: integer

estado:

type: string

metodo_pago:

type: string

direccion_entrega:

type: string

```
fecha_pedido:  
  type: string  
  format: date-time  
  
PedidoCompleto:  
  allOf:  
    - $ref: '#/components/schemas/PedidoOutput'  
    - type: object  
      properties:  
        usuario:  
          $ref: '#/components/schemas/UsuarioPerfil'  
        items:  
          type: array  
          items:  
            type: object  
            properties:  
              producto_titulo:  
                type: string  
              cantidad:  
                type: integer  
              precio_unitario:  
                type: integer  
              subtotal:  
                type: integer  
              historial_estados:  
                type: array
```

items:

type: object

properties:

estado:

type: string

fecha:

type: string

format: date-time

updated_at:

type: string

format: date-time

ComentarioInput:

type: object

required:

- producto_id

- contenido

- calificacion

properties:

producto_id:

type: integer

example: 1

contenido:

type: string

minLength: 5

maxLength: 1000

example: "Excelente calidad, perfecto para mi bebé prematuro"

calificacion:

type: integer

minimum: 1

maximum: 5

example: 5

ComentarioUpdate:

type: object

properties:

contenido:

type: string

minLength: 5

maxLength: 1000

calificacion:

type: integer

minimum: 1

maximum: 5

ComentarioOutput:

type: object

properties:

id:

type: integer

usuario:

\$ref: '#/components/schemas/UsuarioPerfil'

producto_id:

 type: integer

contenido:

 type: string

calificacion:

 type: integer

created_at:

 type: string

 format: date-time

updated_at:

 type: string

 format: date-time

MensajeInput:

type: object

required:

 - destinatario_id

 - contenido

properties:

destinatario_id:

 type: integer

 example: 2

pedido_id:

 type: integer

 example: 1

contenido:

```
type: string  
minLength: 1  
maxLength: 2000  
example: "Hola, ¿este body viene en talla 00?"
```

MensajeOutput:

```
type: object  
properties:  
  id:  
    type: integer  
  pedido_id:  
    type: integer  
  remitente:  
    $ref: '#/components/schemas/UsuarioPerfil'
```

destinatario:

```
    $ref: '#/components/schemas/UsuarioPerfil'
```

contenido:

```
  type: string
```

leido:

```
  type: boolean
```

created_at:

```
  type: string
```

```
  format: date-time
```

Conversacion:

```
type: object
```

properties:

usuario:

\$ref: '#/components/schemas/UsuarioPerfil'

ultimo_mensaje:

\$ref: '#/components/schemas/MensajeOutput'

mensajes_sin_leer:

type: integer

Notificacion:

type: object

properties:

id:

type: integer

tipo:

type: string

enum:

- nuevo_mensaje
- pedido_actualizado
- producto_favorito
- nuevo_comentario
- sistema

contenido:

type: string

metadata:

type: object

leida:

```
    type: boolean

    created_at:
        type: string
        format: date-time
```

ReporteVentas:

```
    type: object

    properties:
        periodo:
            type: object
            properties:
                desde:
                    type: string
                    format: date
                hasta:
                    type: string
                    format: date

        total_ventas:
            type: integer

        total_pedidos:
            type: integer

        ventas_por_dia:
            type: array
            items:
                type: object
                properties:
```

fecha:
 type: string
 format: date

total:
 type: integer

cantidad_pedidos:
 type: integer

productos_mas_vendidos:
 type: array
 items:
 type: object
 properties:
 producto_id:
 type: integer
 producto_titulo:
 type: string
 cantidad_vendida:
 type: integer
 total_recaudado:
 type: integer

EstadisticasVendedor:
 type: object
 properties:
 total_productos:
 type: integer

```
productos_activos:  
    type: integer  
  
total_ventas:  
    type: integer  
  
total_recaudado:  
    type: integer  
  
ventas_ultimos_30_dias:  
    type: integer  
  
productos_mas_vendidos:  
    type: array  
  
    items:  
        type: object  
  
        properties:  
            producto_id:  
                type: integer  
  
            producto_titulo:  
                type: string  
  
            cantidad_vendida:  
                type: integer  
  
calificacion_promedio:  
    type: number  
  
    format: float
```