

Entrega 1 - grupo 4

agosto 2024

Geronimo Correa
Tomás Peña
Felipe Rivero

Listado

Parte 1

- Home
- Detalle producto
- Carrito de compra
- Iniciar sesión
- Registrarse

* Perfil

* Historial de compras

Parte 2

- Navegación

Parte 3

- Listado de dependencias

Parte 4

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

Parte 5

- contrato de datos de la API REST



Detalle producto

Logo

Inicio | Perfil | Iniciar sesión | Registrarse

Carrito

título

img

Descripción y detalles

Añadir al carrito

Footer

Carrito de compra



Iniciar sesión



Registrarse

The wireframe illustrates a registration page layout. It features a dark gray header bar at the top. On the left side of the header is a white box labeled "Logo". In the center of the header is a white box containing the navigation links "Inicio | Perfil | Iniciar sesión | Registrarse". On the right side of the header is a white box labeled "Carrito". The main content area has a light gray background and contains a large, rounded rectangular box in the center labeled "Componente de registro". At the bottom of the page is a solid blue footer bar labeled "Footer".

Perfil

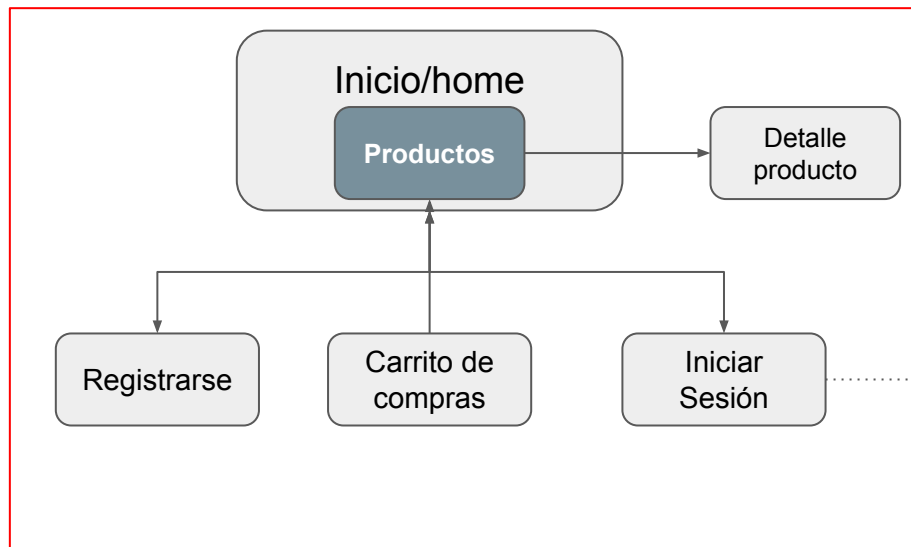


Historial de compras



2. Navegación

sitio público



sitio privado



3. Listado de dependencias

Dependencias Frontend

- Vite
- react
- react-dom
- react-router-dom
- Bootstrap
- Jest
- Axios
- DotEnv

Dependencias Backend

- Express
- Cors
- Morgan
- PG
- DotEnv
- JWT (jsonwebtoken)
- Jest
- Supertest

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

[illegible]

5. contrato de datos de la API REST

1. Registro de Usuario (POST /users)

```
{  
  "email": "usuario@ejemplo.com",  
  "password": "mi_contraseña_segura",  
  "picture": "url_de_la_foto"  
}
```

Response

```
{  
  "id": 1,  
  "email": "usuario@ejemplo.com",  
  "picture": "url_de_la_foto"  
}
```

5. contrato de datos de la API REST

2. Inicio de Sesión (POST /login)

```
{  
  "email": "usuario@ejemplo.com",  
  "password": "mi_contrasena_segura"  
}
```

Response

```
{  
  "token": "jwt_token",  
  "user": {  
    "id": 1,  
    "email": "usuario@ejemplo.com",  
    "picture": "url_de_la_foto"  
  }  
}
```

5. contrato de datos de la API REST

3. Obtener Instrumentos (GET /instruments)

```
[
  {
    "id": 1,
    "name": "Guitarra Stratocaster",
    "price": 500000,
    "category": "Guitarras Eléctricas",
    "description": "Una guitarra eléctrica icónica con un sonido versátil."
  },
  {
    "id": 2,
    "name": "Piano Digital",
    "price": 1200000,
    "category": "Pianos & Teclados",
    "description": "Piano digital con 88 teclas y múltiples opciones de sonido."
  }
]
```

5. contrato de datos de la API REST

4. Añadir Producto al Carrito (POST /cart)

```
{  
  "instrument_id": 1,  
  "quantity": 2  
}
```

Response

```
{  
  "message": "Producto añadido al carrito."  
}
```


5. contrato de datos de la API REST

5. Obtener Productos en el Carrito (GET /cart)

```
[  
  {  
    "instrument_id": 1,  
    "name": "Guitarra Stratocaster",  
    "price": 500000,  
    "quantity": 2  
  },  
  {  
    "instrument_id": 2,  
    "name": "Piano Digital",  
    "price": 1200000,  
    "quantity": 1  
  }  
]
```

5. contrato de datos de la API
REST

6. Eliminar Producto del Carrito (DELETE /cart/instrument_id)

Response

```
{  
  "message": "Producto eliminado del carrito."  
}
```

5. contrato de datos de la API REST

7. Actualizar Cantidad en el Carrito (PATCH /cart/)

```
{  
  "quantity": 3  
}
```

Response

```
{  
  "message": "Cantidad actualizada."  
}
```

5. contrato de datos de la API REST

8. Añadir Nuevo Instrumento (POST /instruments)

```
{  
  "name": "Nuevo Instrumento",  
  "price": 100000,  
  "category": "Nueva Categoría",  
  "description": "Descripción del nuevo instrumento.",  
  "imageUrl": "url_de_la_imagen"  
}
```

Response

```
{  
  "id": 3,  
  "name": "Nuevo Instrumento",  
  "price": 100000,  
  "category": "Nueva Categoría",  
  "description": "Descripción del nuevo instrumento.",  
  "imageUrl": "url_de_la_imagen",  
  "message": "Instrumento añadido con éxito."  
}
```

5. contrato de datos de la API REST

9. Eliminar Instrumento (DELETE /instruments/instrument_id)

Response

```
{  
  "message": "Instrumento eliminado con éxito."  
}
```

5. contrato de datos de la API REST

10. Actualizar Detalles de Instrumento(PATCH/instruments/)

```
{  
  "name": "Instrumento Actualizado",  
  "price": 150000,  
  "category": "Categoría Actualizada",  
  "description": "Descripción actualizada.",  
  "imageUrl": "url_actualizada_de_la_imagen"  
}
```

Response

```
{  
  "message": "Instrumento actualizado con éxito."  
}
```