**GESTOR DE GASTOS**

1. **Introducción**
   1. **Objetivo del sistema**

Este sistema tiene como objetivo proporcionar una herramienta práctica, simple e intuitiva para el manejo de gastos personales. Para ello, se centrará sobre el registro de gastos y proporcionará estadísticas que permitirán al usuario planificar sus finanzas.

* 1. **Funcionalidad del sistema**

El sistema brindará al usuario la capacidad de registrar gastos varios de manera sencilla, permitiéndole escoger entre distintas actividades que haya realizado. Además, organizará dichos registros y le permitirá al usuario ver análisis de sus gastos a través de estadísticas y gráficos comparativos.

1. **Requisitos del Sistema**
   1. **Requisitos Funcionales (RF)**

**Gestión de usuarios**

* **RFa01**:El sistema debe permitir a los usuarios registrarse con un correo electrónico y contraseña.
* **RFa02**: El sistema debe permitir a los usuarios iniciar sesión con sus credenciales.
* **RFa03**: El sistema debe permitir cerrar sesión.
* **RFa04**: Cada usuario debe acceder únicamente a sus propias operaciones y estadísticas.

**Registros de operaciones**

* **RFb01**: El sistema debe permitir al usuario agregar una nueva operación.
* **RFb02**: Al agregar una operación, el usuario debe seleccionar una categoría y subcategoría de entre las opciones predefinidas.
* **RFb03**: Al agregar una operación, el usuario debe seleccionar una cuenta asociada (Efectivo, Crédito o Débito).
* **RFb04**: Al agregar una operación, el usuario debe ingresar la cantidad gastada.
* **RFb05**: El sistema debe permitir al usuario agregar una nota opcional a la operación.
* **RFb06**: El sistema debe registrar automáticamente la fecha de creación de la operación, permitiendo su edición.
* **RFb07**: Las operaciones registradas no podrán ser modificadas una vez guardadas.
* **RFb08**: El sistema debe permitir eliminar operaciones previamente registradas.
* **RFb09**: El campo “Monto” debe aceptar únicamente valores numéricos con separador decima (punto o coma) y hasta dos dígitos decimales.

**Consulta y visualización**

* **RFc01**: El usuario debe poder ver su lista de operaciones registradas, y agrupada por día y mes.
* **RFc02**: El usuario debe poder consultar el detalle completo de cada operación (categoría, cuenta, cantidad, nota, fecha).

**Análisis de gastos**

* **RFc03**: El sistema debe mostrar un gráfico de la tendencia de los gastos diarios del usuario en los últimos 30 días.
* **RFc04**: El sistema debe mostrar un gráfico que compare los gastos del usuario por categoría en los últimos 30 días.
  1. **Requisitos No Funcionales (RNF)**
* **RNF01**: El sistema debe estar implementado como una aplicación web accesible desde navegadores modernos en computadoras de escritorio o portátiles.
* **RNF02**: La aplicación debe requerir autenticación (registro e inicio de sesión) para acceder a las funciones principales.
* **RNF03**: Los datos deben ser almacenados de forma segura en el servidor, separados por cuenta de usuario.
* **RNF04**: La interfaz debe ser intuitiva y optimizada para pantallas de escritorio.
* **RNF05**: El sistema debe encriptar las contraseñas y proteger los datos personales del usuario.
* **RNF06**: Las operaciones del usuario deben estar disponibles desde cualquier dispositivo tras iniciar sesión.
* **RNF07**: El sistema debe guardar automáticamente la fecha actual como predeterminada, si el usuario no indica otra.

1. **Diagramas de Casos de Uso**
   1. **Gestión de Usuario**

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | **Actor** | **Descripción** |
| Registrarse | Usuario | El usuario crea una cuenta con correo electrónico y contraseña. |
| Iniciar sesión | Usuario | El usuario accede a su cuenta con sus credenciales. |
| Cerrar sesión | Usuario | El usuario finaliza su sesión en el sistema. |

* **CU01 – Registrarse**

**Actor**: Usuario  
**Objetivo**: Crear una cuenta personal.  
**Precondición**: El correo de usuario no debe estar registrado.  
**Postcondición**: Se crea una cuenta y el usuario queda autenticado.

**Flujo básico:**

1. El usuario accede a la pantalla de registro.
2. Ingresa un correo electrónico válido y una contraseña.
3. Confirma el registro.
4. El sistema valida y crea la cuenta.
5. El sistema inicia sesión automáticamente o redirige al login.

**Flujos alternativos:**

1. Correo ya registrado: El sistema informa el error.
2. Contraseña no válida (ejemplo: muy corta): el sistema informa y solicita corrección.

* **CU02 – Iniciar sesión**

**Actor**: Usuario  
**Objetivo**: Acceder a la cuenta personal.  
**Precondición**: La cuenta debe estar registrada.  
**Postcondición**: El usuario queda autenticado en el sistema.

**Flujo básico:**

1. El usuario accede a la pantalla de inicio de sesión.
2. Ingresa su correo y contraseña.
3. El sistema valida las credenciales.
4. Si son correctas, el usuario accede a la aplicación.

**Flujos alternativos:**

1. Credenciales incorrectas: mensaje de error
2. Cuenta inexistente: mensaje de error

* **CU03 – Cerrar sesión**

**Actor**: Usuario  
**Objetivo**: Salir de la aplicación.  
**Precondición**: El usuario debe estar autenticado.  
**Postcondición**: Se cierra la sesión y se redirige a la pantalla de inicio/login.

**Flujo básico:**

1. El usuario hace clic en “Cerrar sesión”.
2. El sistema termina la sesión y redirige al inicio.

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | **Actor** | **Descripción** |
| Agregar operación | Usuario | El usuario ingresa una nueva operación de gasto. |
| Eliminar operación | Usuario | El usuario elimina una operación previamente registrada. |
| Ver lista de operaciones | Usuario | El usuario visualiza sus operaciones agrupadas por fecha y ordenadas cronológicamente. |
| Ver detalle de operación | Usuario | El usuario consulta la información completa de una operación. |
| Ver gráfico de tendencias | Usuario | El usuario consulta un gráfico de sus gastos diarios en los últimos 30 días. |
| Ver gráfico por categoría | Usuario | El usuario consulta un gráfico que muestra un total gastado por categoría. |

* 1. **Gestión de Operaciones y Análisis**

ACLARACIÓN: Para esta actividad el usuario deberá haber sido autenticado (iniciado sesión) previamente.

* **CU04 – Agregar operación**

**Actor**: Usuario  
**Objetivo**: Registrar una operación con categoría, método de pago, cantidad y demás datos.  
**Precondición**: La aplicación debe estar abierta y funcional.  
**Postcondición**: La operación queda almacenada y visible en la lista.

**Flujo básico:**

1. El usuario selecciona la opción “Agregar operación”.
2. La aplicación solicita:  
   - Categoría y subcategoría (obligatorio)  
   - Método de pago (obligatorio)  
   - Cantidad (obligatorio)  
   - Nota (opcional)  
   - Fecha (opcional, por defecto utiliza la actual)
3. El usuario confirma el ingreso.
4. El sistema guarda la operación.
5. Se muestra un mensaje de confirmación.

**Flujos alternativos:**

1. El usuario no completa un campo obligatorio: El sistema indica el error y no permite continuar.
2. Falla al guardar por error interno: Se informa al usuario y no se guarda la operación.

* **CU05 – Eliminar operación**

**Actor**: Usuario  
**Objetivo**: Borrar una operación registrada.  
**Precondición**: La operación ya fue creada.  
**Postcondición**: La operación ya no está en el sistema.

**Flujo básico:**

1. El usuario visualiza la lista de operaciones.
2. Selecciona una operación y elige la opción “Eliminar”.
3. El sistema solicita confirmación.
4. El usuario confirma.
5. El sistema elimina la operación y muestra un mensaje de éxito.

**Flujos alternativos:**

1. El usuario cancela la eliminación: No se realiza ningún cambio.

* **CU06 – Ver lista de operaciones**

**Actor**: Usuario  
**Objetivo**: Consultar todas las operaciones registradas.  
**Precondición**:   
**Postcondición**: El sistema muestra la lista de operaciones.

**Flujo básico:**

1. El usuario accede a la sección “Operaciones”.
2. El sistema muestra la lista, ordenada cronológicamente por fecha y agrupada por mes.

**Flujos alternativos:**

1. El listado está vacío: Se muestra un mensaje de lista vacía.

* **CU07 – Ver detalle de operación**

**Actor**: Usuario  
**Objetivo**: Ver información completa de una operación específica.  
**Precondición**:   
**Postcondición**: El usuario accede al detalle de una operación.

**Flujo básico:**

1. El usuario accede a la lista de operaciones.
2. Selecciona una operación.
3. El sistema muestra:  
   - Categoría y subcategoría  
   - Método de pago  
   - Cantidad  
   - Nota (si hay)  
   - Fecha

* **CU08 – Ver gráfico de tendencias**

**Actor**: Usuario  
**Objetivo**: Ver cómo ha evolucionado el gasto en los últimos 30 días.  
**Precondición**:   
**Postcondición**: El sistema muestra el gráfico de tendencias

**Flujo básico:**

1. El usuario accede a la sección de análisis.
2. Selecciona “Tendencia de gasto”.
3. El sistema genera y muestra un gráfico de gasto diario en los últimos 30 días.

**Flujos alternativos:**

1. No se han generado registros: Se muestra un mensaje de lista vacía.

* **CU09 – Ver gráfico por categoría**

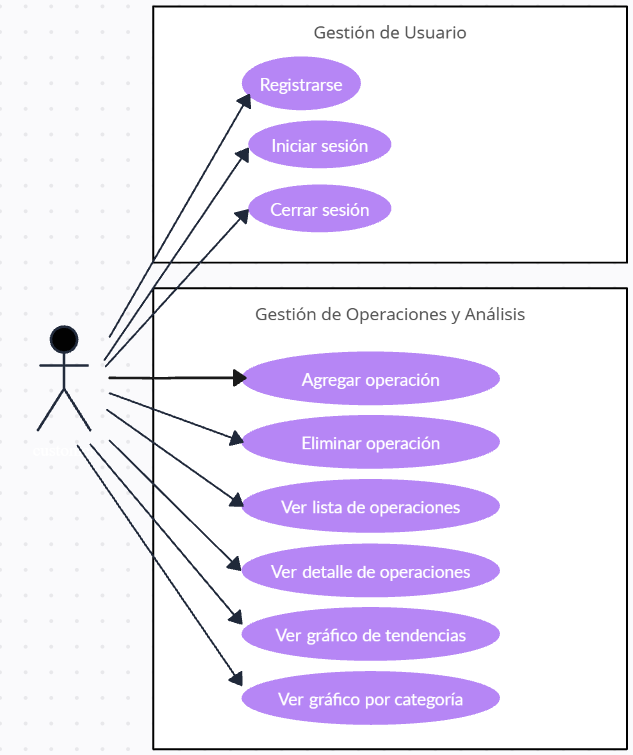
**Actor**: Usuario  
**Objetivo**: Ver cuánto gastó por categoría.  
**Precondición**:   
**Postcondición**: El sistema muestra el gráfico por categoría

**Flujo básico:**

1. El usuario accede a la sección de análisis.
2. Selecciona “Gasto por categoría”.
3. El sistema genera un gráfico con el total gastado por cada categoría en los últimos 30 días.

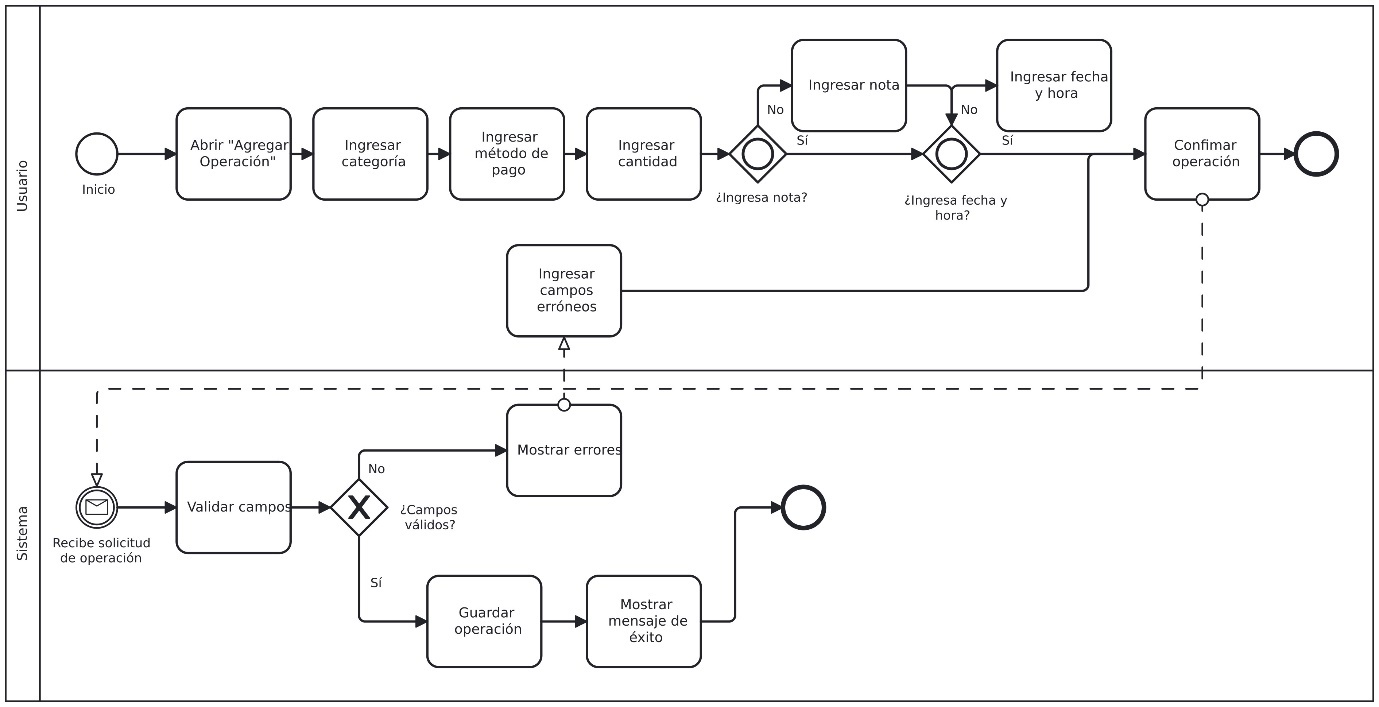
**Flujos alternativos:**

1. No se han generado registros: Se muestra un mensaje de lista vacía.



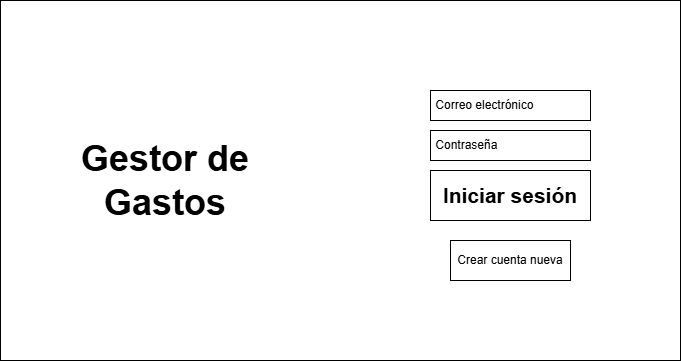
1. **Diagramas de BPMN**
   1. **Proceso CU04 – Agregar operación**

Descripción: Este es uno de los procesos más importantes que suceden en la aplicación. En este proceso el usuario ingresará a la pantalla de “agregar operación” y registrará detalladamente una operación. Una vez confirmados los datos, el sistema validará cada campo ingresado por el usuario y guardará la operación.



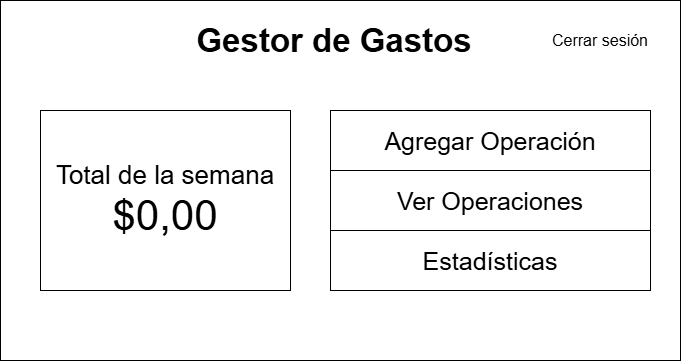
* 1. **Otros procesos**

Descripción: Durante el desarrollo de las siguientes iteraciones se realizarán y detallarán los diagramas BPMN de los procesos restantes.

1. **Prototipado**
   1. **Pantalla de inicio**

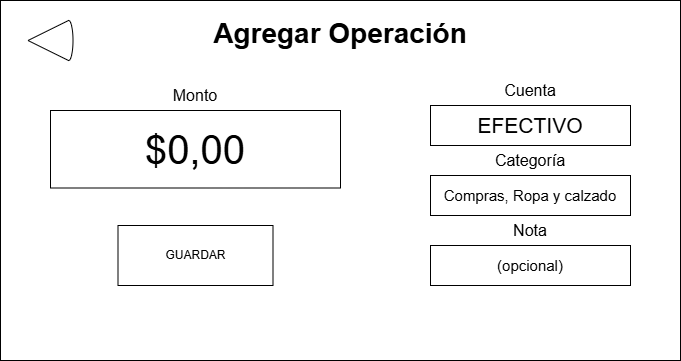
Al ingresar a la página web el usuario se encontrará con la pantalla de inicio. Aquí tendrá dos opciones: Iniciar sesión o registrarse. Si desea iniciar sesión deberá haber registrado una cuenta anteriormente y tendrá que ingresar su correo electrónico y contraseña correctamente.

* 1. **Pantalla principal**

****

Luego de iniciar sesión el usuario accederá a la pantalla principal. Esta contiene un total de gastos de la semana y los ítems: Agregar Operación, Ver Operaciones y Estadísticas. El usuario puede hacer clic en cualquiera de ellos para acceder a sus determinadas pantallas. Además, podrá Cerrar sesión desde dicha pantalla principal cuando desee.

* 1. **Agregar operación**



Si el usuario selecciona el ítem “Agregar Operación” será dirigido a una pantalla para registrar un gasto. En ella tendrá un botón de guardado con la que registrará su operación y además otros ítems con la siguiente información:

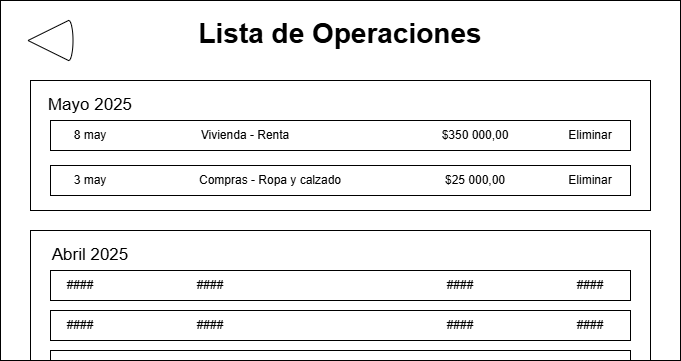
**Monto**: Cantidad de dinero gastada en una sola operación. Se ingresarán números por teclado.

**Cuenta**: Método de pago utilizado. El usuario hará clic en la cacilla y abrirá un menú con diferentes opciones de métodos de pago.

**Categoría**: Tipo de utilidad adquirida. El usuario hará clic en la cacilla y abrirá un menú con diferentes opciones de categoría. Cada categoría a su vez se compone de subcategorías.

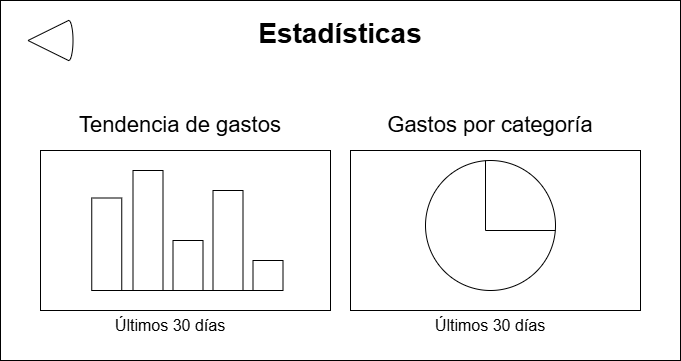
**Nota**: Descripción opcional de la operación. Se ingresará por teclado. El usuario podrá elegir o no si ingresar una descripción que se verá reflejada en su operación guardada.

* 1. **Lista de operaciones**

****

Si el usuario selecciona el ítem “Ver Operaciones” será dirigido a una pantalla que muestra un listado de las operaciones registradas hasta el momento. El usuario podrá desplazarse por la pantalla y hacer clic en cada una de las operaciones para mirarlas en detalle o eliminarlas.

* 1. **Estadísticas**



Si el usuario selecciona el ítem “Estadísticas” será dirigido a una pantalla que muestra el menú de estadísticas de gastos. Tendrá la opción de ingresar a la pantalla de Tendencia de gastos o a la de Gastos por categoría.

1. **Implementación**

La implementación se encuentra en la página pública de GitHub del proyecto: <https://github.com/ManuelMansilla/Gestor_de_gastos>

1. **Casos de Prueba**

La documentación del Test Case o Casos de Prueba está detallada en planillas de Excel adjuntas en la carpeta “Testing”.

(Primera Iteración)  
**ID TC\_001** [Agregar una operación exitosa]: Verificación del registro de una operación exitosa.  
**ID TC\_002** [Verificar monto vacío]: Verificación de la validación correcta del registro al intentar guardar un monto vacío. **ID TC\_003** [Eliminar una operación]: Verificación de funcionamiento de eliminar una operación registrada. **ID TC\_004** [Verificar display responsive]: Verificación de funcionamiento de la tecnología responsive  
  
(Tercera Iteración)

…

1. **Arquitectura**

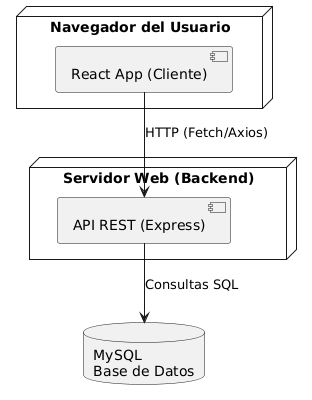
**8.1 Arquitectura Cliente-Servidor**

La arquitectura elegida para el sistema es de Cliente-Servidor. Esta separación permite que el cliente (frontend) interactúe con el servidor (backend) a través de solicitudes HTTP y así se establece una clara división de responsabilidades.

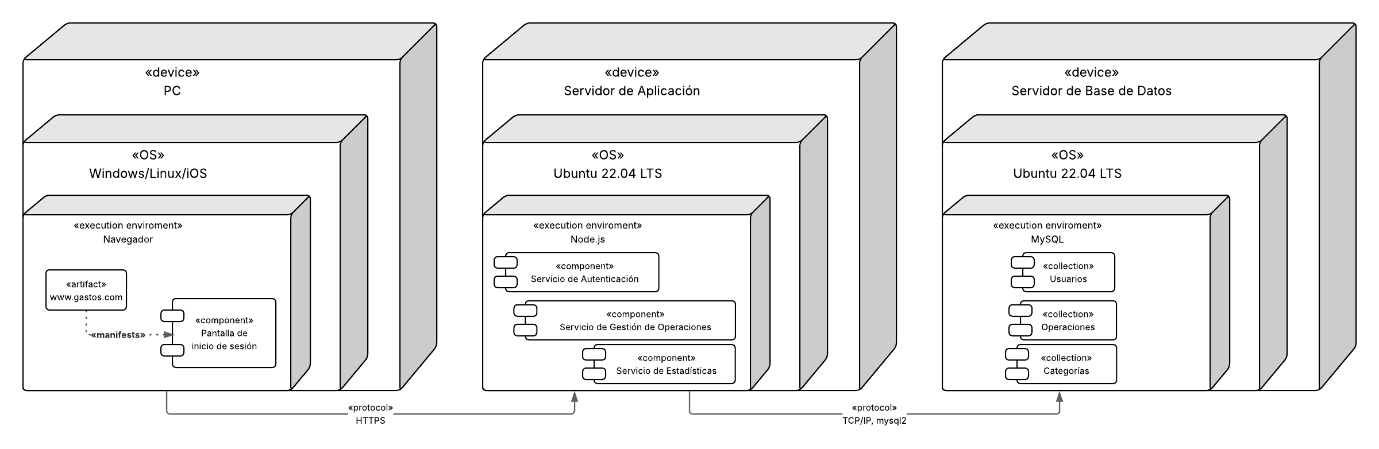
El cliente está desarrollado con React y se ejecuta en el navegador del usuario. Se encarga de mostrar la interfaz gráfica, gestionar la navegación y enviar solicitudes a la API.

El servidor está construido con Node.js y Express. Su función es recibir las solicitudes del cliente, procesarlas, comunicarse con la base de datos MySQL y devolver respuestas adecuadas. Además, gestiona la autenticación de usuarios mediante cookies.

**8.2 Diagramas de Arquitectura**  
  
El siguiente diagrama de despliegue representa cómo se distribuyen los componentes del sistema entre cliente, servidor y base de datos.



El siguiente diagrama de componentes muestra la organización interna del sistema, representando los módulos principales del cliente (desarrollado en React), los servicios lógicos del servidor de aplicación (Node.js + Express) y las colecciones de datos gestionadas por el servidor de base de datos (MySQL). También se indican las relaciones e interacciones entre los diferentes componentes.



1. **Documentación de la API REST**

La API del sistema sigue el estilo REST y utiliza autenticación basada en cookies.

**9.1 Autenticación**● Se utiliza “cookie de sesión” con httpOnly y secure.  
● Los endpoints protegidos requieren sesión iniciada.

**9.2 Endpoints de la API**

**Endpoints de Autenticación**

|  |  |  |
| --- | --- | --- |
| **Método HTTP** | **Endpoint** | **Descripción** |
| **POST** | /api/auth/register | Registra un nuevo usuario. |
| **POST** | /api/auth/login | Inicia sesión y establece una cookie. |
| **POST** | /api/auth/logout | Cierra sesión eliminando la cookie. |

1. Registro de usuario (/api/auth/register)

● Ejemplo JSON:

|  |
| --- |
| {  "email": "usuario@ejemplo.com",  "password": "\*\*\*\*\*\*\*\*"  } |

● Respuestas:

- 201 Created: Usuario creado.

- 400 Bad Request: Email ya registrado.

1. Inicio de sesión (/api/auth/login)

● Ejemplo JSON:

|  |
| --- |
| {  "email": "usuario@ejemplo.com",  "password": "\*\*\*\*\*\*\*\*"  } |

● Respuestas:

- 200 OK: Sesión iniciada.

- 401 Unauthorized: Credenciales inválidas.

1. Cierre de sesión (/api/auth/logout)

● Respuestas:

- 200 OK: Sesión finalizada.

**Endpoints de Operaciones**

|  |  |  |
| --- | --- | --- |
| **Método HTTP** | **Endpoint** | **Descripción** |
| **GET** | /api/operations | Lista todas las operaciones del usuario autenticado. |
| **POST** | /api/operations | Crea una nueva operación. |
| **DELETE** | /api/operations/:id | Elimina una operación específica |

1. Lista de operaciones (/api/operations)

● Ejemplo JSON:

|  |
| --- |
| {  {  “id”: 1,  “categoria”: “Compras”,  “subcategoria”: “Ropa”,  “cuenta”: “Efectivo”,  “monto”: 53.00,  “fecha”: “2025-04-02T12:00:00Z”,  “nota”: “Remera nueva”  }  } |

1. Registrar operación (/api/operations)

● Ejemplo JSON:

|  |
| --- |
| {  {  “categoria”: “Compras”,  “subcategoria”: “Ropa”,  “cuenta”: “Efectivo”,  “monto”: 53.00,  “fecha”: “2025-04-02T12:00:00Z”,  “nota”: “Remera nueva”  }  } |

● Respuestas:

- 201 Created: Operación registrada.

- 400 Bad Request: Faltan datos.

1. Eliminar operación (/api/operations/:id)

● Respuestas:  
- 200 OK: Operación eliminada exitosamente.

- 404 Not Found: Operación no encontrada.

**Endpoints de Estadísticas**

|  |  |  |
| --- | --- | --- |
| **Método HTTP** | **Endpoint** | **Descripción** |
| **GET** | /api/stats/overview | Devuelve resumen de gastos (semana,mes). |
| **GET** | /api/stats/trend | Devuelve gastos diarios de los últimos 30 días. |
| **GET** | /api/stats/categories | Devuelve gasto por categoría. |

1. Resumen de gastos (/api/stats/overview)
2. Tendencia de gastos de los últimos 30 días (/api/stats/trend)
3. Gastos comparados por categoría (/api/stats/categories)