

**PROYECTO DE INTEGRACIÓN**

**DISEÑO DE UNA APLICACIÓN WEB PARA GESTIÓN DE GASTOS  
COMPARTIDOS**

**JUAN FERNANDO SAENZ S.  
JUAN SEBASTIAN ROMERO  
MAGDALENA TRIVINO**

**PRESENTADO A:  
LAURA PATRICIA MANCERA VALETTS**

**UNIVERSIDAD EAN  
FACULTAD DE INGENIERÍA**

**BOGOTÁ, JUNIO 1 DE 2022**

## TABLA DE CONTENIDO

<i>Índice de figuras</i> .....	7
<b>1. RESUMEN EJECUTIVO</b> .....	<b>8</b>
<b>2. INTRODUCCIÓN</b> .....	<b>9</b>
<b>3. OBJETIVOS</b> .....	<b>11</b>
<b>4. DEFINICIÓN DEL PROBLEMA</b> .....	<b>12</b>
<b>5. JUSTIFICACIÓN</b> .....	<b>13</b>
<b>6. ANÁLISIS DE REQUERIMIENTOS</b> .....	<b>15</b>
<b>6.1 Alcance del proyecto</b> .....	<b>15</b>
<b>6.2 Características del producto</b> .....	<b>15</b>
<b>6.3 Definición de requerimientos y arquitectura de negocio</b> .....	<b>16</b>
6.3.1 Proceso de levantamiento .....	16
6.3.2 Definición de requerimientos .....	17
<b>6.4 Casos de uso</b> .....	<b>18</b>
<b>6.5 Clasificación de requerimientos</b> .....	<b>25</b>
<b>6.6 Propuesta de Interfaz Gráfica de Usuario</b> .....	<b>29</b>
<b>7. BENCHMARKING DE APLICACIONES</b> .....	<b>33</b>
<b>8. MARCO TEÓRICO</b> .....	<b>40</b>

<b>8.1. Aplicaciones Web.....</b>	<b>40</b>
<b>9. GLOSARIO DE TÉRMINOS.....</b>	<b>47</b>
<b>10. DISEÑO METODOLOGICO.....</b>	<b>48</b>
<b>11. ANÁLISIS DE RESTRICCIONES Y LMITACIONES.....</b>	<b>51</b>
<b>12. ANÁLISIS DE ALTERNATIVAS .....</b>	<b>54</b>
<b>13. ANÁLISIS DE COSTOS.....</b>	<b>59</b>
<b>17. REFORMULACION INTERFAZ GRAFICA DE USUARIO .....</b>	<b>64</b>
<b>18. PLAN DE IMPLEMENTACIÓN .....</b>	<b>68</b>
<b>19. SOLUCION DE INGENIERIA .....</b>	<b>75</b>
<b>20. OPCIONES DE MEJORA Y TRABAJOS FUTUROS.....</b>	<b>75</b>
<b>21. CONCLUSIONES .....</b>	<b>90</b>
<b>22. LISTA DE REFERENCIAS .....</b>	<b>91</b>

## Índice de tablas

<i>Tabla 1. Caso de uso para registrar un usuario.....</i>	<i>20</i>
<i>Tabla 2 Caso uso de la aplicación web para grupos. ....</i>	<i>21</i>
<i>Tabla 3.....</i>	<i>21</i>
<i>Tabla 4 Caso para consultar el balance en la aplicación web.....</i>	<i>22</i>
<i>Tabla 5 Caso para edición de perfil en aplicación web.....</i>	<i>23</i>
<i>Tabla 6 Caso de inicio de sesión en aplicación web.....</i>	<i>24</i>
<i>Tabla 7 Clasificación de los requerimientos.....</i>	<i>25</i>
<i>Tabla 8 convenciones para benchmarking de aplicaciones.....</i>	<i>33</i>
<i>Tabla 9 benchmarking de aplicaciones .....</i>	<i>34</i>
<i>Tabla 10 Comparación de alternativas comerciales de aplicaciones para control de gastos grupales .....</i>	<i>37</i>
<i>Tabla 11. Alternativas para desarrollo de aplicación.....</i>	<i>54</i>
<i>Tabla 12. Sistema de calificación para evaluación de alternativas .....</i>	<i>55</i>
<i>Tabla 13. Evaluación de alternativas para desarrollo de la aplicación. ....</i>	<i>56</i>
<i>Tabla 14. Análisis de variables.....</i>	<i>60</i>
<i>Tabla 15. Tabla de costos directos .....</i>	<i>62</i>

***Tabla 16. Tabla de costos indirectos ..... 62***

***Tabla 17. Tabla de costos total..... 63***

## Tabla de ilustraciones

Ilustración 1. Pantalla de login .....	29
Ilustración 2. Menú inicio.....	30
Ilustración 3. Pantalla de resultados .....	31
Ilustración 4. Pantalla de inicio .....	64
Ilustración 5. Registro de usuario .....	64
Ilustración 6. Pantalla registro de usuario .....	65
Ilustración 7. Registro de planes.....	65
Ilustración 8. Pantalla de planeación .....	66
Ilustración 9. Pantalla de interacción.....	66
Ilustración 10. Pantalla de resultados .....	67
Ilustración 11. Detalles de balance .....	67
Ilustración 12. Estructura del proyecto.....	67
Ilustración 13. Estructura del objeto de base de datos.....	67
Ilustración 14. Estructura del enrutador .....	78
Ilustración 15. Estructura de pagina principal de la aplicacion .....	79
Ilustración 16. Estructura del backend local en construccion .....	80
Ilustración 17. Estructura del frontend basado en componentes .....	82
Ilustración 18. Estructura de un componente reutilizable del frontend .....	67
Ilustración 19. Pantalla de inicio de sesion de la aplicacion .....	67

Ilustración 20. Pantalla principal de usuario de la aplicación ..... 87

Ilustración 21. Pantalla de detalles de un plan..... 88

### **Índice de figuras**

*Figura 1 Flujo de creación de usuario.* ..... 19

## **1. RESUMEN EJECUTIVO**

Las asociaciones de gastos compartidos pueden aplicarse en muchas áreas diferentes, desde las salidas casuales con amigos hasta la renta. La contabilidad y el reparto de gastos pueden complicarse rápidamente más aún cuando no se cuenta con una organización. Para acometer el problema se plantea el diseño de una aplicación web la cual permita inteligentemente la división de dichos gastos, esto a través de un sistema de contabilidad que simplifica el proceso.

A pesar de su pertinencia actual y amplia disponibilidad de aplicaciones, hasta el momento se han encontrado tan solo algunas aplicaciones web y trabajos de carácter académico sobre la problemática de reparto de gastos. Los objetivos de este proyecto son, por un lado, consultar y diseñar métodos adecuados de reparto de gastos y, por otro, calcular con exactitud el total a pagar de cada miembro de un grupo. Como conclusión se espera compilar dichas funcionalidades en una aplicación web de fácil uso e intuitiva.



## 2. INTRODUCCIÓN

Hoy en día es común que un grupo de amigos o colegas realicen actividades que los llevan a compartir gastos. Este acuerdo suele requerir una cuidadosa negociación que exige confianza y toca temas relacionados con la igualdad, las normas y expectativas sociales. Por este motivo, la división de las facturas o los gastos compartidos puede tener una ecuación diferente, dependiendo del tipo de relación. La gestión de las finanzas compartidas implica intereses individuales y comunes. Estas negociaciones pueden dar lugar a valoraciones de equidad que influyen en la calidad de la relación.

Normalmente, la división de los gastos en grupos ya sea en un viaje o simplemente en una salida con amigos, compartir la cuenta puede ser complicado. ¿Se divide el coste en partes iguales o desglosa las bebidas y los platos principales de cada uno? ¿Cuánto debe aportar cada persona para la propina? Y dividir las facturas más grandes -como la renta, los servicios públicos y el mantenimiento de un apartamento- puede añadir aún más complejidad.

Si bien los aspectos prácticos son evidentes, estos no se escapan del interés académico: publicaciones en revistas de matemáticas aplicadas orientadas a la solución de gastos compartidos en viajes, así como tesis de maestría enfocadas en el desarrollo de software pertinente, una de las cuales brinda el sustento científico a una de las aplicaciones más populares en la tienda de aplicaciones móviles de Android.

A pesar de la popularidad del tema, de las múltiples opciones disponibles en el mercado y de la variedad de opciones que cada una de estas brinda, se identifican debilidades tales como la

necesidad de instalación local, la disponibilidad exclusiva para un mismo sistema operativo y el requisito de crear una cuenta para poder hacer uso de los servicios.

Tras identificar estas necesidades, surge la iniciativa del presente proyecto en el cual nos proponemos diseñar una aplicación web que permita dicha gestión de manera eficiente, fácil y evitando malentendidos, permitiendo a cada participante conocer su balance y sugiriendo los movimientos a realizar para saldar las cuentas de todo el grupo, presentando una alternativa a las aplicaciones móviles mediante la creación de una aplicación web pensada en las pantallas de estos dispositivos, que no requiera realizar instalación y sea independiente del sistema operativo.

### **3. OBJETIVOS**

#### **3.1. OBJETIVO GENERAL**

Desarrollar una herramienta que permita la correcta y ágil gestión de los gastos compartidos por medio de un aplicativo Web que facilite el proceso de cálculo y balance transaccional.

#### **3.2. OBJETIVOS ESPECÍFICOS**

- Organizar los datos de las entidades que interactúan en la aplicación y asegurar su persistencia, mediante el diseño de una base de datos.
- Especificar la estructura, funcionamiento e interacción de los componentes mediante el diseño de la arquitectura de la aplicación.
- Elaborar un prototipo funcional por medio del desarrollo de los módulos de la aplicación.
- Asegurar el adecuado funcionamiento de la aplicación mediante el desarrollo de pruebas unitarias.
- Agregar a disposición de los usuarios la aplicación por medio del despliegue de la aplicación en un servicio de hosting.

#### **4. DEFINICIÓN DEL PROBLEMA**

Llevar el registro de los gastos personales es una recomendación que se suele recibir en contextos de planeación financiera. El cumplirlo de manera disciplinada podría resultar tedioso y, dependiendo de la necesidad y compromiso personal, puede terminar inconcluso si adicionalmente no se plantean metas claras. Quizás las consecuencias no vayan más allá de los desbalances y pérdidas usuales a las cuales se está acostumbrado.

Sin embargo, en el momento de llevar a cabo actividades grupales surge la necesidad de incluir actores adicionales no solo en el registro, sino en el pago de gastos compartidos en escenarios como contribuciones no equitativas entre participantes, inclusión de consumidores sin responsabilidad de pago, pagos de una misma cuenta con aportes parciales, préstamos entre participantes, entre otras.

En el nuevo escenario los cálculos de balances para cada participante incrementan su complejidad, al igual que la determinación de transacciones para saldar deudas entre participantes, lo cual suele generar desigualdad en los pagos, pérdidas, sobrecostos por transacciones innecesarias, deudas y malentendidos.

Teniendo en cuenta lo mencionado se plantea la pregunta de investigación: ¿De qué manera puede optimizarse la gestión de gastos compartidos por un grupo de personas, para lograr una gestión ágil de los cálculos y balances transaccionales?

## 5. JUSTIFICACIÓN

La necesidad de tener claridad en la gestión de los gastos grupales se alimenta de la realidad del manejo y seguimiento de las cuentas en escenarios reales con su complejidad creciente respecto del número de participantes y transacciones realizadas. Eventos como salidas a comer con amigos, viajes grupales, pago de servicios del hogar de manera contributiva, entre otros, son ejemplos de situaciones que implican el compartir gastos con personas. Debido a razones que incluyen la no disponibilidad de medios de pago y carencia de alternativas, consumos no homogéneos, acuerdos de pago no equitativos, gastos no planeados, etc., surge la fuente de desbalances en las cuentas para cada participante, así como errores en futuros pagos y repercusiones tanto a nivel económico como personal.

Las situaciones planteadas no representan un caso aislado: se encuentran contempladas en escenarios académicos que promueven soluciones en el entorno de las matemáticas aplicadas a gastos compartidos en viajes (Koman & Ticha, 1998) como en tesis de maestría (Vavra, 2012), (Parra et al, 2019) que abordan directamente el problema de compartir gastos y establecer balances, por medio del desarrollo de aplicaciones móviles.

Adicionalmente, el interés por gestionar los gastos compartidos no se restringe al ámbito académico. Basta una revisión de las tiendas de aplicaciones para descubrir la oferta para dispositivos móviles (Billr, s.f.), (Cloud Height Technologies Limited, 2022), (Vavra, 2022), (Split, s.f.), (SplitWise, s.f.), (Steffgriffy, s.f.), (Tricount, 2022), así como aplicaciones web

(Kitty Split, 2022), (Tricount, 2022), (BillZer, s.f.) que proponen soluciones similares al mismo problema base.

La posibilidad de contar con herramientas para llevar un control de gastos grupales se orienta primordialmente hacia la división exacta de las obligaciones de pago pudiendo garantizar una repartición justa en el momento de saldar cuentas y realizar pagos. Se destaca además la posibilidad de cruzar cuentas de modo que se reduzca el número de transacciones que pueden tener asociados costos además de requerir tiempo extra. De los beneficios expuestos se derivan consecuencias favorables adicionales con respecto a factores personales y de relacionamiento, tanto con miembros del grupo como con terceros responsables de la facturación en establecimientos.

La mayoría de las herramientas existentes si bien se enfocan en los dispositivos que son mayormente utilizados en la actualidad, requieren de una instalación local y creación de cuenta. Algunas son de pago, lo cual, a pesar de garantizar una mayor calidad, restringe a los usuarios que requieren darle a la aplicación un uso ágil e inmediato.

Por lo anterior, se propone la elaboración de una aplicación web que sea accesible desde cualquier dispositivo sin necesidad de realizar instalación, y que se presente como alternativa a las no tan numerosas opciones que presentan esta característica, teniendo la posibilidad de crear una cuenta en caso de requerir.

## **6. ANÁLISIS DE REQUERIMIENTOS**

### **6.1 Alcance del proyecto**

El objetivo de la aplicación web para la gestión de gastos compartidos es facilitar la gestión del reparto de gastos y crear una aplicación cómoda y fácil de usar para los usuarios, que quieren mantener un orden en sus finanzas compartidas. El sistema se basará en una base de datos relacional con sus funciones de gestión y añadir pagos/deudas. Se dispondrá de un servidor de base de datos, así como la conversión a diversas monedas. Sobre todo, se espera ofrecer una experiencia de usuario cómoda e intuitiva.

### **6.2 Características del producto**

Los usuarios del sistema deben poder recuperar de la base de datos la información de sus gastos compartidos. El sistema soportará un tipo de privilegios de usuario. Los usuarios tendrán acceso a las funciones de usuario. El usuario deberá poder realizar las siguientes funciones:

## **6.3 Definición de requerimientos y arquitectura de negocio.**

### **6.3.1 Proceso de levantamiento**

Para el proceso de levantamiento de requerimientos se tiene en cuenta un instrumento de recolección de datos tipo Encuesta, el cual consta de preguntas abiertas para evitar poner limitantes a las personas participantes y conocer de manera más personal sus intenciones en torno a la aplicación.

- ¿Has sufrido algún tipo de malentendido con tus amigos, familiares o pareja al momento de dividir los gastos en los viajes o salidas comunes?
- ¿Qué tipo de malentendido has sufrido por dividir los gastos o por no controlar tus gastos?
- ¿Conoces alguna aplicación o página web que se dedique a dividir los gastos en las salidas de viajes o cotidianas con tus amigos o familiares?
- ¿Qué mecanismos o herramientas tecnológicas usas al momento de compartir los gastos?
- ¿Llevas un registro de los gastos asumidos de tu parte y tus amigos en los viajes o salidas comunes?
- ¿Te gustaría crear un bote común virtual desde el que se puedan hacer pagos?
- ¿Pagarías más de \$20.000 mensuales por servicios premium como exportación de archivos de tus cuentas de gastos compartidos, evitar anuncios y una mejor experiencia en aplicaciones de gastos compartidos?



### **6.3.2 Definición de requerimientos**

Los requerimientos funcionales son aquellos que indican lo que debe permitir el sistema realizar al usuario, es decir los servicios que el sistema debería prestar al usuario (Pantaleo y Rinaudo, 2015). A continuación, se definen los requerimientos funcionales.

1. Definir un grupo de personas para la división de gastos.
2. Cada grupo deber ser asignado con un nombre
3. Se debe insertar la cantidad de personas para la división
4. Se debe insertar el nombre de las personas
  - 4.1 Solo una persona debe estar con la sesión iniciada en la aplicación.
5. Añadir, editar y eliminar miembros de un grupo
6. Añadir los pagos: quién pagó, para quién, cuánto, con qué propósito.
7. Registro de pagos de cada persona.
8. Registro de pagos total.
9. Liquidar/calcular deudas.
10. Marcar la deuda como liquidada/pagada.
11. Soporte para múltiples monedas y conversión automática a la moneda del grupo
12. Reparto desigual o igual dependiendo de selección.
13. El algoritmo de liquidación de deudas

Los requerimientos no funcionales se refieren a cómo debería funcionar un sistema, se refieren a las restricciones y cualidades del sistema (Pantaleo y Rinaudo, 2015). Los requerimientos no funcionales se listan a continuación.

1. Ante una eventual caída el sistema realizará una copia de seguridad en menos de 10 horas.
2. El sistema podrá procesar 100 transacciones por segundo
3. Cualquier consulta a la base de datos no tardará más de 3 segundos
4. El sistema realizará una copia de seguridad cada 48 horas, la cuál será enviada a un servidor
5. El sistema validará la identificación (email) del usuario al ingresar al mismo.
6. Para la encriptación de contraseñas se usará base64
7. La aplicación tendrá un menú de ayuda con explicaciones para los usuarios.
8. La interfaz tendrá un diseño responsive
9. La aplicación estará disponible las 24 horas del día, los 7 días de la semana; excepto durante periodos de mantenimiento programado.
10. El sistema estará disponible en español con posibilidad de traducir la página

#### **6.4 Casos de uso.**

A continuación, mostraremos los casos de usos para este proyecto.

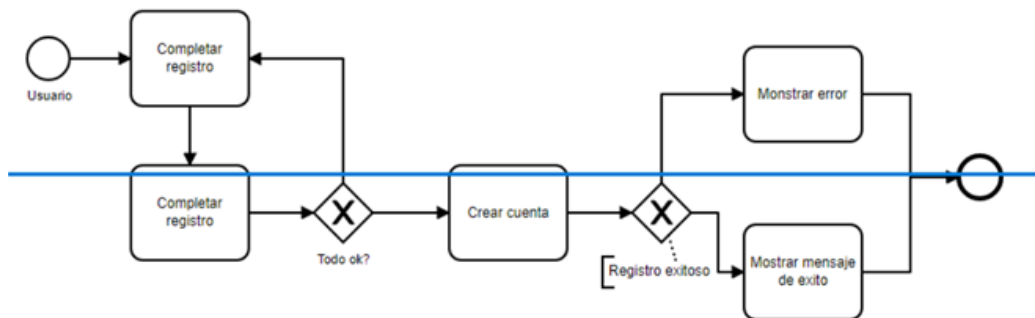
#### **Registro de usuarios.**

El registro puede ser realizado por un usuario invitado cuando un usuario accede a nuestro sitio web.

Al registrar una cuenta, para simplificar el proceso de registro, sólo se solicita la información necesaria del usuario. Lo único que se solicita es: correo electrónico, nombre completo y contraseña. Este número será el nombre de usuario que se utilizará para acceder a la página web.

El flujo de información para el registro de usuarios se ilustra en la figura 1. Los casos de uso relacionados tanto con la creación del usuario cómo agregar grupos, deudas/gastos, consultar balances, editar perfiles de usuario, así como la gestión de inicio de sesión, se encuentran descritos en las tablas 1 a la 6.

*Figura 1 Flujo de creación de usuario.*



*Fuente: Elaboración propia*

**Tabla 1. Caso de uso para registrar un usuario.**

<b>Caso de uso 1</b>	<b>Registrar un usuario</b>
Actores	Usuario
Descripción	Registra una cuenta con un rol de Usuario que permitirá todas las diferentes funcionalidades de los usuarios
Condición previa	Acceder a un formulario de registro de un sitio web
Flujo básico	<ol style="list-style-type: none"><li>1. Rellenar toda la información requerida (nombre, correo electrónico, contraseña)</li><li>2. Hacer clic en el botón "Enviar"</li><li>3. Confirmar un correo electrónico haciendo clic en un enlace enviado a su bandeja de entrada</li></ol>
Flujo alternativo	Si alguno de los campos obligatorios no se ha rellenado, muestra un mensaje de error sin permitirle completar su registro.
Condición posterior	Un usuario se registrará en un sitio web y podrá acceder a su cuenta

*Fuente: Elaboración propia*

**Tabla 2 Caso uso de la aplicación web para grupos.**

<b>Caso de uso 2</b>	<b>Agregar grupo</b>
Actores	Usuario
Descripción	Permite agregar un grupo de amigos, colegas/ personas para la división de un gasto compartido.
Condición previa	Se ha registrado el usuario y ha accedido a la página "grupos".
Flujo básico	<ol style="list-style-type: none"><li>1. Escribir nombre del grupo y cantidad de personas.</li><li>2. Digitar el nombre de cada persona del grupo.</li><li>3. Hacer clic en "guardar"</li></ol>
Flujo alternativo	Si no se rellenan algunos campos obligatorios (nombre de grupo o nombre de usuarios), muestra un mensaje de error.
Condición posterior	Se añadirá un grupo en la base de datos

*Fuente: Elaboración propia*

**Tabla 3 Caso para agregar el tipo de procedimiento requerido en la aplicación web**

<b>Caso de uso 3</b>	<b>Agregar deuda/gasto.</b>
----------------------	-----------------------------

Actores	Usuario
Descripción	Permite agregar una deuda/gasto.
Condición previa	Se ha registrado el usuario y ha accedido a la página "gastos".
Flujo básico	<ol style="list-style-type: none"> <li>1. Haga clic en "agregar gasto"</li> <li>2. Ingrese monto</li> <li>3. Confirme haciendo clic en sí o en no</li> </ol>
Flujo alternativo	Si hubo un error al agregar el gasto en la base de datos, muéstrelo.
Condición posterior	La información de una cita será eliminada de la base de datos.

*Fuente: Elaboración propia*

**Tabla 4 Caso para consultar el balance en la aplicación web**

<b>Caso de uso 4</b>	<b>Consultar balance</b>
Actores	Usuario
Descripción	Permite mostrar el balance (pagos totales) del grupo.
Condición previa	Se ha registrado el usuario y ha accedido a la página "gastos".

Flujo básico	<ol style="list-style-type: none"> <li>1. Haga clic en "gastos"</li> <li>2. El balance total se mostrará en la parte izquierda.</li> </ol>
Flujo alternativo	N/A
Condición posterior	El resultado se mostrará en una página.

*Fuente: Elaboración propia*

**Tabla 5 Caso para edición de perfil en aplicación web**

<b>Caso de uso 5</b>	<b>Editar perfil</b>
Actores	Usuario
Descripción	Permite editar datos del usuario
Condición previa	Se ha registrado el usuario y ha accedido a la página "Editar perfil".
Flujo básico	<ol style="list-style-type: none"> <li>1. Elige el módulo perfil</li> <li>2. Podrá editar nombre de usuario, contraseña y email.</li> <li>3. Haga clic en el botón "Actualizar".</li> </ol>
Flujo alternativo	Si uno de los campos obligatorios no se rellena, mostrará un error al hacer clic en el botón " Actualizar".

Condición posterior	El usuario será actualizado en la una base de datos. Ahora se puede visualizar en la página "Mi perfil".
---------------------	--

*Fuente: Elaboración propia*

**Tabla 6 Caso de inicio de sesión en aplicación web.**

<b>Caso de uso 6</b>	<b>Inicio de sesión</b>
Actores	Visitantes
Descripción	Permite acceder el sitio web
Condición previa	Acceder a una página de "Login/Registro" en el sitio web.
Flujo básico	<ol style="list-style-type: none"> <li>1. Rellene el nombre de usuario y la contraseña.</li> <li>2. Haga clic en el botón "Login"</li> </ol>
Flujo alternativo	<p>Si uno de los campos obligatorios no se rellena, muestra un error al hacer clic en el botón "Login" de la página web.</p> <p>Si el nombre de usuario y/o la contraseña no son correctos, mostrar un mensaje de error.</p>



Condición posterior	Un usuario se registrará en el sitio web. Podrá utilizar las funcionalidades de ese rol en particular.
---------------------	--

*Fuente: Elaboración propia*

## 6.5 Clasificación de requerimientos

Establecer la prioridad de los requerimientos es muy importante para realizar la planificación del proyecto (Pantaleo y Rinaudo, 2015).

**Tabla 7 Clasificación de los requerimientos**

Nombre del Requerimiento	Descripción específica del Requerimiento	Tipo de Requerimiento	Prioridad del Requerimiento
Gestión de usuario	El usuario iniciará sesión y tendrá acceso a la creación de grupos y posteriormente con el grupo creado, podrá agregar gastos/ deudas.	Funcional	Alta

Inicio de sesión de usuario	Los usuarios accederán al sistema autenticandose. Y tendrá un rol predefinido.	Funcional	Alta
Gestión de grupos.	El usuario podrá consultar sus grupos y ver su historial de gastos en cada grupo.	Funcional	Media
Copia de seguridad	En menos de 5 horas el sistema realizará una copia de seguridad de la información del sistema, se guardará en un servidor alternativo. El back up se realizará cada 48 horas	No funcional	Alta
Diseño del sistema	El sistema será responsivo, pudiendo acceder al mismo a través de cualquier dispositivo móvil. Tendrá fondo blanco y letras negras para facilitar la comodidad de la lectura.  Igualmente, los usuarios podrán	No funcional	Baja

	acceder a un menú de ayuda. El menú estará disponible en español con posibilidad de traducir la página.		
Transacciones	El sistema podrá procesar 100 transacciones por segundo. Cualquier consulta a la base de datos no tardará más de 3 segundos.	No funcional	Alta
Disponibilidad	La aplicación estará disponible todo el tiempo (excepto durante los periodos de mantenimiento del mismo).	No funcional	Alta
Validaciones	El sistema valida la identificación del usuario cuando este ingresa.	No funcional	Alta
Seguridad	Las contraseñas de los usuarios están encriptadas por	No funcional	Alta

	ley, se usará base64 para garantizar esto.		
Usabilidad	El tiempo de aprendizaje del administrador será inferior a 10 minuto ya que el sistema es muy intuitivo.	No funcional	Baja

*Fuente: Elaboración propia*

## 6.6 Propuesta de Interfaz Gráfica de Usuario.

- Pantalla de login

En la pantalla de ingreso damos inicio a el programa en el cual los usuarios ya registrados ingresaran a la aplicación y los nuevos usuarios se pueden registrar de manera gratuita o como invitado.

*Ilustración 1. Pantalla de login*



*Fuente: Elaboración propia mediante mockups.*

- Pantalla de inicio:

Una vez el usuario o invitado haya ingresado aparecerá la pantalla inicial, en la cual se debe ingresar el número de personas y nombre de cada una de ellas, así mismo también escoger el tipo de evento que es: viaje, comida o facturas. También se pueden crear varios planes y varios grupos de personas.

*Ilustración 2. Menú inicio*

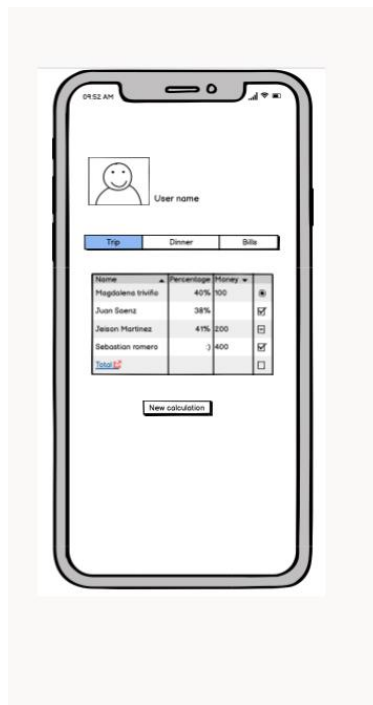


*Fuente: Elaboración propia mediante mockups.*

- Pantalla de resultados:

La pantalla de resultados muestra el tipo de evento también un cuadro en el cual se puede ver los nombres de cada persona con el porcentaje y el valor a pagar individualmente también permite compartir los resultados con las personas que uno desee y realizar nuevos cálculos.

*Ilustración 3. Pantalla de resultados*



*Fuente: Elaboración propia mediante mockups*

La aplicación contará con las siguientes características:

- Disponible para IOS y Android
- Gratuita para invitados
- Lista de costos por persona
- Capacidad para grupos de varias personas
- Posibilidad de compartir los resultados con las personas que desees
- Reportes y porcentaje por persona
- Soporta varios planes



## 7. BENCHMARKING DE APLICACIONES

Con el fin de revisar el estado actual en cuanto a las aplicaciones disponibles en el mercado para control de gastos compartidos, se realiza una búsqueda en línea en las tiendas disponibles tanto para Android como para iOS.

Las aplicaciones más populares que se encontraron se listan a continuación, junto a una convención para evaluación posterior:

**Tabla 8 convenciones para benchmarking de aplicaciones**

<b>Aplicación</b>	<b>Convención</b>
Billr.me	A
Billzer	B
Group Expenses	C
KittySplit	D
Settle Up	E
Split	F
Sesterce	G

Splitwise	H
Tricount	I
Bluecoins	J

Fuente: Elaboración propia

**Tabla 9 benchmarking de aplicaciones**

APLICACIÓN	A	B	C	D	E	F	G	H	I	J
<i>Libre de cargos</i>		x				x	x	x	x	
<i>Móvil Android</i>					x	x	x	x	x	x
<i>Móvil iOS</i>	x				x					
<i>Web</i>		x	x	x	x					
<i>Uso sin registro</i>		x		x		x	x		x	x
<i>Invitados sin registro</i>		x		x			x	x		x
<i>Distribución de cuentas no equitativa</i>			x	x	x			x	x	
<i>Soporta varios planes/proyectos</i>			x				x	x	x	
<i>Lectura automática de facturas</i>	x								x	x

<i>Conexión con bancos</i>			x							
<i>Notificaciones por email/SMS</i>	x		x			x	x			x
<i>Participantes ilimitados</i>						x	x			x
<i>Aportes por más de una persona</i>	x									x
<i>Genera reportes</i>			x	x			x			
<i>Triangulación de cuentas</i>				x	x	x		x	x	
<i>Disponible en Colombia</i>	x	x	x	x	x	x	x	x	x	x
<i>Enlaces compartidos</i>					x		x		x	x
<i>Versiones premium</i>					x		x	x	x	x
<i>Exportación de archivos (PDF. CSV)</i>									x	x
<i>Cambio de moneda</i>			x	x		x	x			x

*Fuente: Elaboración propia*

Se evaluaron 20 ítems de las aplicaciones y páginas web más notables e influyentes del mercado en torno a gastos compartidos y páginas web, comparando sus diversas funcionalidades, características y usos que la misma aplicación ofrece a sus usuarios. De esta manera se busca determinar y comparar qué aplicaciones tienen funciones similares y cuáles no, para de esta manera realizar una aplicación más eficiente, funcional y práctica.

Basado en el estudio de benchmarking, se puede evidenciar que las aplicaciones de SettleUp, Bluecoins, Splitwise y Group Expense presentan una gran cantidad de alternativas de manejo desde su aplicación, presentando de esta manera una gama alta de funciones prácticas para sus usuarios, entre las cuales podemos encontrar la exportación de archivos pdf, alianzas con bancos, recordatorios, resúmenes de presupuestos y artículos, interfaces sencillas y prácticas, multifuncional entre diversos usuarios, su funcionalidad a nivel de cuentas económicas es amplia, presentado porcentajes y gráficos al instante, además de presentar etiquetas de uso, para una mejor experiencia de los usuarios. Por otro lado, las aplicaciones de Split y Bilzer son las más básicas respecto a las investigadas, estas presentan una interfaz sencilla, pero no cuentan con funciones avanzadas de filtros, no generan gran interacción entre diversos usuarios y su uso está limitado a funciones básicas de cuentas sencillas.

La mayoría de las aplicaciones presentan una versión premium de pago, las cuales traen funciones extras en las mismos, como lo son: desbloqueo de anuncios, mensajes de textos bancarios, sincronización en línea, mayor almacenamiento de información financiera, proyecciones futuras, tablas y gráficos, los precios de las versiones premium varían dependiendo de la aplicación que se usará, teniendo un costo desde \$ 23.500 pesos hasta \$ 99.000 pesos colombianos.

Se presenta a continuación una relación de características de las opciones comerciales para aplicaciones de gestión de gastos grupales.

**Tabla 10 Comparación de alternativas comerciales de aplicaciones para control de gastos grupales**

	SPLITWISE	SETTLE UP	SPLIT	TRICOUNT	BLUECOINS
Definición	Administrador de gastos gratuitos para hacer cuentas entre amigos de viaje	Aplicación de registro de gastos Gestiona los gastos	Gestiona los gastos compartidos, entre amigos, parejas o compañeros	Divide gastos y cuentas de grupo	Aplicación de finanzas simple que da seguimiento a los gastos, presupuestos y administrador de dinero
Versión Gratuita	Crear grupos Liquidar deudas Pizarra Saludos	Crea grupos Búsqueda de grupos cercanos Enlaces de invitación	Nuevos gastos Nuevo pago Nueva persona	Gastos Saludos Creación de grupos función de realizar domicilios	Transacciones Recordatorios Hoja de balance Ganancia neta Resumen de presupuesto Resumen de artículos

Versión Premium	Escanea recibos Libre de anuncios Conversión de monedas Desglose de elementos Tablas y gráficos Búsqueda de gastos	Sin anuncios Categorías de gastos Añadir fotos de recibos Recordar pagos Transacciones recurrentes y futuras	Sin versión premium	Sin anuncios en la aplicación Estadísticas Divide gastos en partes desiguales Exportación a formatos PDF o CSV Asistencia técnica	Sin anuncios Sincronización en línea Proyección futura Filtro avanzado Mensajes de texto bancarios
Valor Premium	99.900,00/año	33.500,00/año	-----	31.540,26/año o	23.500,00/año
Etiquetas	Viaje, proyectos, otros	Gastos compartidos	Sin etiquetas	Viaje, casa compartida, pareja, celebración,	Casa, automóvil, entretenimiento, utilidades, empleador y otros

				proyecto, otros	
Inicio De Sesión	Facebook  Google	Facebook  Google	-----  -	Facebook  Google	-----
Fecha De Lanzamiento	12 sept 2011	19 abr 2011	20 NOV  2017	1 jul 2011	2 abr 2016
Compatible	SO Android  7.0 y versiones posteriores	SO Android  6.0 y versiones posteriores	SO Android  4.4 y versiones posteriores	SO Android  5.0 y versiones posteriores	SO Android  4.1 y versiones posteriores
Descarga Desde Play Store	10,000.000	1,000.000	500.000	1,000.000	1,000.000

*Fuente: Elaboración propia*

## **8. MARCO TEÓRICO**

Resulta pertinente revisar algunos conceptos con respecto a las temáticas directamente relacionadas con el proyecto, entre los que se encuentran los de Aplicaciones Web por ser el tipo de aplicación que se contempla, el Modelo Cliente – Servidor que define la responsabilidad de los macroelementos de la aplicación, el Patrón MVC como uno de los patrones estructurales más utilizado y de particular interés, así como los algoritmos para balance de cuentas (de elaboración propia) y de cruce de cuentas para realizar los pagos efectivos reduciendo número de transacciones.

### **8.1. Aplicaciones Web.**

Una aplicación Web es un sitio Web que contiene páginas con contenido sin determinar, parcialmente o en su totalidad. El contenido final de una página se determina sólo cuando el usuario solicita una página del servidor Web. Dado que el contenido final de la página varía de una petición a otra en función de las acciones del visitante, este tipo de página se denomina página dinámica. Las aplicaciones Web se crean en respuesta a diversas necesidades o problemas.



## **8.2. Modelo Cliente - Servidor.**

La arquitectura cliente – servidor es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien da la respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajoso en un sistema operativo multiusuario distribuido a través de una red de computadoras (Coa, 2013)

La solución planteada combina sistemas de gestión de información, constituyendo una aplicación web, basada desarrollada sobre plataformas digitales de programación, La aplicación posee un diseño sencillo y práctico, además de una interfaz práctica, la cual hará de la experiencia de los usuarios más cómoda, facilitando su uso y comprensión de esta. Adicionalmente se permite el almacenamiento, actualización y eliminación de información financiera pertinente al usuario y el ofrecer los posibles impactos financieros que tuvo la realización de una compra o gastos compartidos. Las diversas cuentas de gastos compartidos que pueda crear un usuario serán almacenadas para su posterior retroalimentación del usuario para poder identificar posibles errores en las cuantías financieras, mostrando estos resultados de una forma organizada y comprensible para el posterior análisis

Como mecanismos de seguridad y control de acceso, el sistema contara con encriptación de la contraseña de usuarios y formulario de autenticación, además de contar con 10 requerimientos para su pleno funcionamiento.

### 8.3. Patrón MVC.

El patrón MVC es una propuesta de arquitectura del software utilizada para separar el código por sus distintas responsabilidades, manteniendo distintas capas que se encargan de hacer una tarea muy concreta, y el cual ofrece la creación de aplicativos de mayor calidad. (Ye-jin, 2021)

Es por ello por lo que para el desarrollo de nuestra aplicación web tendremos en cuenta las siguientes capas:

- **Modelos:** Es la capa donde se trabaja con los datos, por tanto, contendrá mecanismos para acceder a la información y también para actualizar su estado. Los datos los tendremos habitualmente en una base de datos, por lo que en los modelos tendremos todas las funciones que accederán a las tablas y harán los correspondientes selects, updates, inserts, etc. (Alvarez, 2020)
- **Vistas:** contienen el código de nuestra aplicación que va a producir la visualización de las interfaces de usuario, o sea, el código que nos permitirá renderizar los estados de nuestra aplicación en HTML. En las vistas nada más tenemos los códigos HTML y PHP que nos permite mostrar la salida. En la vista generalmente trabajamos con los datos, sin embargo, no se realiza un acceso directo a éstos. Las vistas requerirán los datos a los modelos y de ellas se generará la salida, tal como nuestra aplicación requiera. (Alvarez, 2020)

- **Control:** Contiene el código necesario para responder a las acciones que se solicitan en la aplicación, como visualizar un elemento, realizar una compra, una búsqueda de información, etc. es una capa que sirve de enlace entre las vistas y los modelos, respondiendo a los mecanismos que puedan requerirse para implementar las necesidades de nuestra aplicación. Sin embargo, su responsabilidad no es manipular directamente datos, ni mostrar ningún tipo de salida, sino servir de enlace entre los modelos y las vistas para implementar las diversas necesidades del desarrollo. (Alvarez, 2020)

### **8.1. Algoritmo para balance de cuentas.**

Para llevar a cabo el balance de cada uno de los miembros de un grupo, se deben tener en cuenta las siguientes variables:

- Plan (PL): cada viaje, salida, proyecto o similar que contemple una serie de Transacciones.
- Persona (PE): cada uno de los miembros participantes de un Plan, los cuales para cada gasto realizan Pagos y definen Compromisos para cada una de las Transacciones.
- Pago (PA): cada contribución individual en dinero que realiza una o varias Personas para cubrir una Transacción.

- Compromiso (CO): valor numérico que refleja la proporción con la una Persona se compromete a cubrir un gasto. Puede ser en porcentaje, en partes, en fracciones o en dinero.
- Transacción (TR): Cada uno de los gastos que se llevan a cabo por los participantes durante un Plan.
- Balance (BA): saldo de cuenta de cada una de las Personas involucradas.

Se tienen las siguientes condiciones y restricciones:

- La suma de todos los pagos de una transacción no puede exceder el monto de la transacción.
- Todos los compromisos de una misma transacción deben tener la misma unidad de medida.
- Para los balances, el saldo acreedor se considera positivo y el saldo deudor se considera negativo.
- La suma de los balances para todos los participantes debe ser igual a cero.

Una vez aseguradas las condiciones, se procede a realizar los cálculos de la siguiente manera:

1. Se suma la totalidad de los Pagos de un Plan: PA-TOT.
2. Se suma la totalidad de Compromisos de un Plan: CO-TOT.
3. Se encuentra la relación entre pago y compromiso totales de un plan:

$$RPACOT = PA-TOT / CO-TOT.$$

4. Para cada transacción, el balance de cada persona se encuentra restando el producto entre el Compromiso y la relación entre pago y compromiso totales de un plan, del Pago realizado:

$$BA = PA - CA * RPACOT$$

5. El balance total para cada persona dentro del plan se calcula como la suma de todos los balances de la persona: BA-TOT.

### **8.1. Algoritmo para cruce de cuentas.**

El algoritmo debe permitir la mejor forma para saldar cuentas entre los participantes, dados los balances de todos (Tricount, 2012). Según menciona el creador de la aplicación Settle Up (Vavra, 2012), para su solución se debe centrar la atención en los saldos en lugar de en los pagos, siguiendo la convención de saldos positivos acreedores y saldos negativos deudores.

“Ordene los miembros según su saldo. El miembro "más rico" se empareja con el "más pobre" y la transacción se realiza entre ellos. Siempre uno o ambos extremos con saldo cero pueden quedar fuera de los cálculos posteriores. El algoritmo se lanza recursivamente en el subconjunto de miembros. Cuando se liquidan todas las deudas, el algoritmo finaliza. El algoritmo producirá en el peor de los casos N-1 transacciones (N es

el número de miembros). También minimiza la cantidad total de dinero transferido, a diferencia de otros enfoques (como encadenar miembros sin clasificar).” (Vavra, 2012)

Aunque se considera un algoritmo adecuado, presenta debilidades cuando existen miembros que no participan en el grupo, ya que sus saldos son cero y podrían excluirse para optimizar. El problema lejos de ser trivial es del tipo polinómico no-determinista (NP) y puede abordarse desde el “problema de la suma de subconjuntos” (Cormen, 2001).

## 9. GLOSARIO DE TÉRMINOS

- Algoritmos: Conjunto de coordenadas o instrucciones definidas, que permiten la solución de un problema.
- Cuentas: Conjunto de cifras y datos de operaciones económicas en una entidad.
- Aplicaciones: Programa informático diseñado como herramienta para realizar una función específica.
- Usuarios: Persona que usa habitualmente un servicio.
- Gestión: Acción o trámite que busca resolver un problema.
- Control: Proceso que verifica el rendimiento mediante comparación con los estándares establecidos
- Pago: Acción que se realiza para extinguir o cancelar una obligación.
- Web: Conjunto de información que se encuentra en una dirección determinada de internet
- Contabilidad: Es la recopilación, registro y la interpretación continuos y sistemáticos de información determinada
- Servidor: Sistema informático que almacena, distribuye y suministra información
- Balances financieros: Es un informe contable que refleja la situación económica y financiera de una persona o empresa
- Transacciones: Acuerdo, comunicación o movimiento llevado a cabo entre un comprador y un vendedor en la que se intercambia un activo contra un pago.

## **10. DISEÑO METODOLOGICO**

A continuación, se detallará la metodología de investigación, las técnicas y herramientas que se implementaron para la recolección de datos, fuentes de información y los resultados obtenidos, para obtener resultados claros y precisos, es necesario aplicar los tipos de investigación, ya que con ayuda de estos se obtendrá resultados concisos y poder cumplir con los objetivos planteados de este proyecto

### **Tipos de investigación**

- Investigación bibliográfica: Es uno de los pilares fundamentales para la obtención de datos sobre otras investigaciones ya existentes tanto en hipótesis, experimentos y técnicas que ayudaran a la búsqueda de información que nos oriente para encontrar la solución a esta investigación
- Investigación cuantitativa: Procedimiento de decisiones que pretende señalar varias alternativas, utilizando las herramientas de recolección de datos como las encuestas, que mediante cálculos estadísticos podemos obtener resultados favorables para la investigación.

### **Métodos de investigación**

Este trabajo adopta una construcción teórica, ya que se implementará la observación para abordar la problemática y de esta manera evaluando los tiempos podemos comprobar si con la



implementación de una herramienta tecnológica habrá mejora en los procesos de gastos compartidos

- Método deductivo: Se refiere al método que va de lo general a lo específico, dando apertura a que los datos generales sean aceptados como válidos, este método conlleva a tener resultados muy buenos en una investigación cuantitativa, por tal razón se aplicará en este proyecto

### **Herramientas de recolección de datos**

- Encuestas: Es una herramienta de recolección de datos, utilizadas para obtener información de personas sobre diversos temas o de uno específico, basados en un grupo de preguntas orientadas a los usuarios que les afecta la problemática de este tema de investigación por resolver. La finalidad de esta herramienta es obtener la información cuantitativa de la población que ha sido encuestada, para que el investigador utilice cualquier método estadístico de tal manera que se pueda medir el interés de las personas y poder ver si las expectativas son las esperadas o no.

- Fuentes primarias: Las fuentes primarias que se implementaron fueron las que se relacionan directamente con los involucrados a quienes les afectan la problemática, empleando la técnica de recolección de información como es la encuesta.

- Fuentes secundarias: son la información que anteriormente fueron recogidas y elaboradas como: datos estadísticos, estudios realizados, bibliografías, que son datos necesarios para el desarrollo de este trabajo

## **11. ANÁLISIS DE RESTRICCIONES Y LMITACIONES**

Resulta pertinente abordar las posibles limitantes y restricciones durante el desarrollo del proyecto, de modo que se puedan orientar los esfuerzos hacia la reducción de alternativas.

Los aspectos a considerar se plantean a continuación:

### **Aspectos ambientales:**

El impacto ambiental esperado para el desarrollo del proyecto se encuentra en el uso de la energía eléctrica. Se destina tiempo considerable frente a los equipos de cómputo y elementos adicionales como iluminación, por lo que el consumo de energía está siempre presente. Para menguar el impacto ambiental, se hará uso de un uso racional de la energía, teniendo los equipos en funcionamiento en los momentos operativos, reduciendo periodos de inactividad.

### **Aspectos económicos:**

Los recursos económicos para llevar a cabo el proyecto se encuentran actualmente disponibles y su uso no presenta una restricción. Se cuenta con los equipos de cómputo necesarios y el

proyecto busca utilizar tecnologías de desarrollo abierto que no impliquen costos adicionales por licencias de uso. Existe cobertura de red y disponibilidad de energía.

#### **Aspectos legales:**

El proyecto hará uso de tecnologías libres, por lo que se espera no se presenten inconvenientes de tipo legal por mal uso de licencias o piratería. En caso de usar elementos tomados de fuentes externas con derechos de autor, se hará la respectiva mención.

#### **Aspectos de salud y seguridad:**

Las posibles alternativas al proyecto comparten los mismos temas de salud y seguridad relacionados con la ergonomía, el manejo del estrés y la jornada de trabajo. Se piensa controlar dichos aspectos con base en la aplicación de buenas prácticas de ergonomía, pausas activas y búsqueda de un balance entre vida personal y actividades de desarrollo del proyecto, de modo que no se presenten inconvenientes. Con respecto a usuarios finales, un posible impacto en la seguridad será la exposición de su teléfono celular en sitios públicos que puedan implicar oportunidad para la delincuencia común. Existen medidas de prevención que los usuarios pueden tomar y coinciden con las recomendaciones de cuidado de objetos personales que los establecimientos presentan.

**Aspectos Socioculturales:**

No se esperan cambios significativos en los hábitos de los usuarios.

## 12. ANÁLISIS DE ALTERNATIVAS

Para el proceso de desarrollo de la aplicación de acuerdo con la especificación de requerimientos y el análisis de restricciones, el punto de divergencia principal se encuentra en la selección entre construir una aplicación móvil y una aplicación web.

**Tabla 11. Alternativas para desarrollo de aplicación.**

Alternativa 1	Aplicación móvil
Alternativa 2	Aplicación web

Fuente: Elaboración propia.

Los aspectos relevantes para evaluar son la seguridad, el diseño, el costo, la cobertura de dispositivos, dificultad en el desarrollo y las opciones para despliegue. Cada uno de los aspectos se asocia con una premisa que describe brevemente la futura aplicación.

Para ponderar las calificaciones con respecto a los aspectos a evaluar se define una serie de pesos (porcentajes) en los cuales el de mayor valor corresponde al costo para desarrollar la aplicación dado que es un parámetro determinante para la toma de decisiones.

Se utiliza un sistema de calificaciones que puede tomar los valores entre 1 y 5, descritos en la tabla a continuación:

**Tabla 12. Sistema de calificación para evaluación de alternativas**

Calificaciones	Valor
Mayor impacto en tiempo y costos	1
Impacto medio en tiempo y costos	3
Menor impacto en tiempos y costos	5

Fuente: Elaboración propia.

Teniendo en cuenta los parámetros mencionados previamente, se realiza la evaluación de las alternativas asignando calificación a las premisas y calculando el total ponderado para cada una de ellas, como se ilustra en la tabla:

**Tabla 13. Evaluación de alternativas para desarrollo de la aplicación.**

Aspecto para evaluar	Peso	Aplicación móvil		Aplicación web	
		Premisa	Calif / 5	Premisa	Calif / 5
Seguridad	20%	Se cuenta seguridad (verificación de email y encriptación de contraseñas), además de un modo de recuperación de cuenta.	3	Se cuenta con seguridad (verificación de email, conexión con servidores externos para logeo como Facebook, Google, encriptación de contraseñas)	3
Diseño	20%	La aplicación debería hacer un uso óptimo de los recursos de software y hardware del dispositivo.	3	La aplicación debería hacer un uso óptimo de las opciones del navegador Web, el DOM y responsive design.	4
Costo	30%	Se cuenta con licencias y simuladores disponibles, o bien con los dispositivos físicos para instalar la aplicación.	3	Se cuenta con opciones gratuitas de herramientas como IDE y Hosting	5
Cobertura de dispositivos	10%	Se debe cubrir la mayor parte de dispositivos móviles del mercado, tanto en Android como en iOS.	3	Se debe contar con que todo dispositivo tenga instalado un navegador Web	5



Dificultad en desarrollo	10%	El equipo de diseño y desarrollo debe contar con la experiencia y conocimiento en desarrollo móvil.	1	El equipo de diseño y desarrollo debe contar con experiencia y conocimiento en desarrollo web.	4
Opciones para despliegue	10%	Se cuenta con los dispositivos en físico o simuladores disponibles	3	Se cuenta con opciones de capacidad suficiente para alojar la aplicación.	4
	100%		2,8		4,2

Se puede observar que de acuerdo con los criterios establecidos la alternativa de la aplicación web tiene una mayor puntuación que la alternativa de la aplicación móvil. Las razones se fundamentan principalmente en el costo (premisa de mayor peso y calificación), dado que para la construcción y prueba de la aplicación web actualmente se cuenta con herramientas gratuitas, a comparación con las herramientas de desarrollo móvil y pruebas en plataforma móvil, que pueden tener costos asociados.

Otro aspecto relevante que contribuye a la selección de la alternativa web es cobertura de dispositivos. Mientras la oferta de navegadores es limitada y usualmente en dispositivos móviles se suele contar con la versión más reciente (las tiendas de aplicaciones gestionan las actualizaciones), la oferta de dispositivos móviles es numerosa, las actualizaciones de hardware inducen obsolescencia en dispositivos de modelos anteriores, junto a la diversidad de versiones y

proveedores de sistemas operativos, lo cual dificulta establecer uno o algunos dispositivos y con esto asegurar una cobertura amplia en el público.

La dificultad en desarrollo y despliegue es también un elemento a tener en cuenta, dado que los desarrollos para móvil suelen tener una mayor dificultad intrínseca que las aplicaciones web. En dispositivos móviles se tiene la opción de integrar sensores y piezas de hardware, más para el presente proyecto dichas opciones no representan ventajas al buscar la agilidad, dada la formación actual que posee el equipo de desarrollo.

### **13. ANÁLISIS DE COSTOS**

Del mismo modo en que el trabajo de desarrollo de software tiene implícita una complejidad considerable, el realizar estimaciones de costos tiene la misma cualidad. La metodología utilizada, la disponibilidad de recursos, el manejo de imprevistos, la curva de aprendizaje, la experiencia de los miembros del equipo, el error humano, y aspectos transversales como reparaciones de equipos, derechos de uso y reproducción de materiales, variación en los requerimientos y consumo energético, son algunos de los factores que contribuyen a la dificultad en la estimación.

Trabajos como el de Roque (Roque, 2022) presentan una variedad de métodos de análisis de costos para proyectos de desarrollo de software como COCOMO, COCOMO II, Líneas de código y puntos de función, y las técnicas Delphi. Estos métodos se basan en el tamaño del proyecto, la estimación de esfuerzo de desarrollo y tienen en cuenta metodologías ágiles.

Para el presente trabajo se tomarán algunos elementos limitados, dado que no se utiliza una implementación formal de metodologías ágiles y el equipo de desarrollo cuenta con limitaciones en cantidad de desarrolladores disponibles, por lo que el alcance del proyecto en el tiempo establecido es pequeño.

De este modo, se incluye en el análisis de costos:

14. Costos directos: mano de obra (variable).

15. Costos indirectos: arriendo de lugar de trabajo (fijo), sostenimiento de desarrolladores (variable), consumo energético (variable), acceso a Internet (fijo), capacitaciones (variable).

16. Capital de trabajo: Equipo de cómputo (fijo).

Dada la posibilidad de realizar trabajo de manera remota, así como el tiempo límite para desarrollar el proyecto y sabiendo que se trata de un intangible, se pueden agrupar los costos de manera que los costos variables queden incluidos dentro de una tarifa que incluya cobertura por alimentación y conectividad, junto a un salario para cada desarrollador (se cuenta con dos desarrolladores en el proyecto).

**Tabla 14. Análisis de variables**

Variables	SI	NO
1. Costo de programación	x	
2. Acceso a datos de la aplicación	x	
3. Geoposicionamiento para la aplicación	x	
4. Realidad aumentada para la aplicación		x
5. Complejidad de la aplicación		x
6. Gestor de contenidos de la aplicación	x	

7. Pasarela de pago para la aplicación	x	
8. registro de usuarios en la aplicación	x	
9. Envío de notificaciones push		X
10. Diseño gráfico para la aplicación	x	
11. el n de plataformas donde será accesible	x	
12. Integración con otros sistemas	x	

Fuente: Elaboración propia.

#### Costos directos:

- Programación (programados por horas)
- Diseño (diseño de la aplicación web)
- Marketing (publicidad en la cual se dará a conocer la aplicación)
- Acceso a datos de la aplicación (acceso a la nube para almacenar los datos de los clientes)

#### Costos indirectos:

- Estudio de mercado (estudio para poder conocer el mercado de ventas)
- Internet (gasto para desarrollar la aplicación)
- Gasto energético (gasto por uso de aparatos tecnológicos)
- Licencia de Microsoft office (licencia para uso de Excel, Word y más)

- Licencia Google (licencia para poder tener una aplicación en la web)

**Tabla 15. Tabla de costos directos**

COSTO DIRECTO					
CONCEPTO	UNIDAD	CANTIDAD	PRECIO UNITARIO	IMPORTE	
programador	pago mensual	1	\$	3.500.000	\$ 3.500.000
diseño	pago mensual	1	\$	500.000	\$ 500.000
marketing	pago mensual	1	\$	2.000.000	\$ 2.000.000
<b>Total, costo directo unitario</b>					\$ 6.000.000
<b>costo indirecto 30%</b>					\$ 1.800.000
<b>costo unitario</b>					\$ 7.800.000

Fuente: Elaboración propia

En el costo directo tenemos en cuenta el proceso que se necesita para la creación de nuestra aplicación web en el cual está el programador, la persona encargada del diseño web y el marketing se calcula el valor unitario mensual para cada uno.

**Tabla 16. Tabla de costos indirectos**

COSTO INDIRECTO					
CONCEPTO	UNIDAD	CANTIDAD	PRECIO UNITARIO	IMPORTE	
Estudio de mercado	pago mensual	1	\$	5.000.000	\$ 5.000.000
Internet	pago mensual	1	\$	100.000	\$ 100.000
Gasto energético	pago mensual	1	\$	80.000	\$ 80.000
licencia Microsoft office	pago mensual	1	\$	90.000	\$ 90.000
Mantenimiento	pago mensual	1	\$	1.200.000	\$ 1.200.000
Licencia Google	pago mensual	1	\$	516.000	\$ 516.000
<b>Costo directo</b>					\$ 6.986.000

<b>Costo directo promedio</b>	
<b>producción</b>	\$ 23.000.000
<b>Costo indirecto</b>	30%

Fuente: Elaboración propia.

En los costos directos se tiene en cuenta los procesos de uso de la aplicación, ya que para el uso de esta aplicación hay unos conceptos los cuales se debe cancelar mensualmente como licencias para el uso de la página web.

**Tabla 17. Tabla de costos total**

COSTO TOTAL	
CONCEPTO	IMPORTE
volumen de producción	3,833333333
costo directo	\$ 23.000.000
costo indirecto	\$ 6.986.000
<b>costo total</b>	<b>\$ 29.986.000</b>

Fuente: Elaboración propia.

El volumen de producción lo calculamos de la división de los costos directos de promedio producción con los costos totales directos y el costo total se calcula de la suma de los costos directos con los costos indirectos.

## 17. REFORMULACION INTERFAZ GRAFICA DE USUARIO

Ilustración 4. Pantalla de inicio

Gestion de Gastos Compartidos

E-mail

Contraseña

Iniciar sesion

No tienes una cuenta?  
[Registrate](#)

### Pantalla inicio de sesión

En esta interfaz se dará el inicio a nuestra sesión de una manera simple y práctica, simplemente ingresaremos nuestro e-mail registrado y la contraseña que ingresamos al momento de crear la cuenta en la plataforma, además de una opción de registrarse por si aún no se tiene una cuenta creada

Ilustración 5. Registro de usuario

Nombre de usuario

E-mail

Contraseña

Repetir contraseña

Registrar Volver

### Pantalla de registro

En la siguiente interfaz nos pedirá el nombre de usuario que deseemos poner, nuestro e-mail personal, la creación de una contraseña, acompañada de una verificación o repetición de la contraseña, y una opción de registrar, para de esta manera crear un usuario



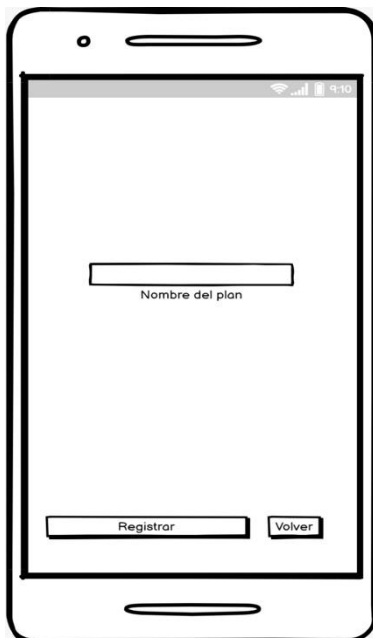
*Ilustración 6. Pantalla registro de usuario*



Pantalla de registro de persona

En esta opción encontramos el registro del nombre de las personas que estarán involucradas en la realización del plan.

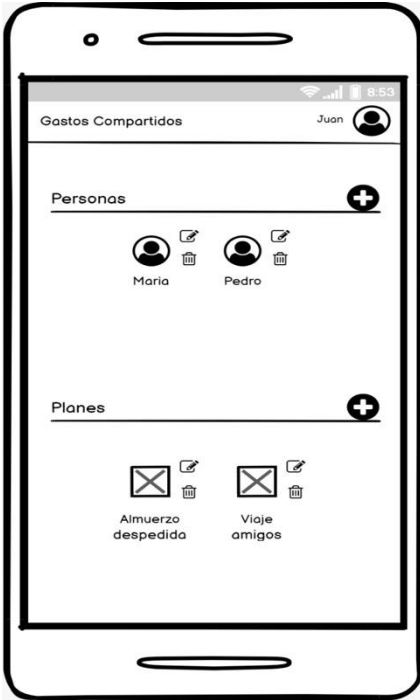
*Ilustración 7. Registro de planes*



Pantalla de registro de plan

En esta opción se encuentra el registro del nombre del plan que se llevara a cabo, los cuales podrán variar dependiendo de la ocasión (cenas, almuerzos, viajes, juegos)

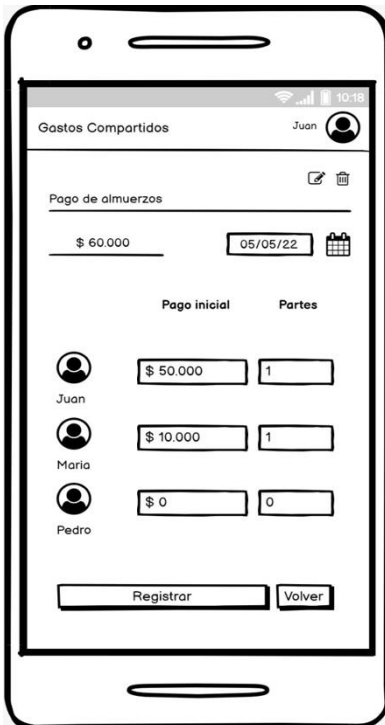
Ilustración 8. Pantalla de planeación



### Pantalla de planeación

En esta opción se encuentran la planeación del plan a llevar a cabo, además de agregar a las personas que participaran en el mismo, con sus respectivas opciones de editar y borrar

Ilustración 9. Pantalla de interacción



### Pantalla de interacción

En esta interfaz se planificará a más profundidad los gastos generados por un tipo de plan y la cantidad monetaria correspondiente de cada participante, además de generar la fecha en la que se realizaron los gastos y la opción de registrar esta transacción

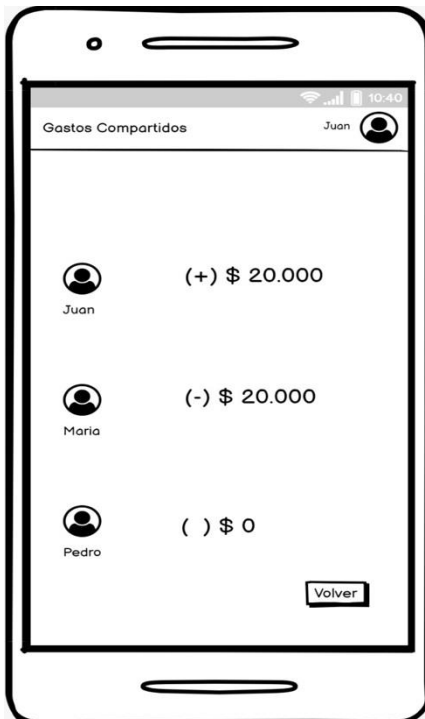
Ilustración 10. Pantalla de resultados



### Pantalla de resultados

En esta última interfaz, se podrá observar las transacciones totales y el respectivo valor de cada una, para el plan que se llevó a cabo, además de arrojar una opción de “ver balance”, con el cual podremos conocer a más detalle la situación económica y financiera del plan.

Ilustración 11. Detalles de balance



### Pantalla detalle de balance

En esta interfaz podremos realizar todo lo relacionado con el monto económico que cada persona pagara respectivamente

## 18. PLAN DE IMPLEMENTACIÓN

Debido a cambios en la estructura del equipo de desarrollo, se hace necesario ejecutar un plan de contingencia que permita contar con un producto mínimo viable en un plazo reducido, de modo que para elaborar el plan de implementación se construye una lista de tareas a realizar, teniendo en cuenta el análisis de requerimientos, aunque manejando una metodología diferente: el desarrollo en cascada.

Para esta metodología se ejecutan de manera **secuencial** los pasos del ciclo de vida del desarrollo de software: Planeación, Análisis, Diseño, Implementación, Pruebas, Despliegue y Mantenimiento. A continuación, se mencionan los cambios a realizar para cada etapa:

### **Etapa de Planeación**

Se decide replantear el alcance del proyecto de modo tal que cuente con las funcionalidades principales en cuanto a lógica de negocio e identificando formas de generar valor a pesar de las modificaciones reduccionistas que se deben hacer.

Las tareas asociadas a la planeación son:

- Redefinir alcance del proyecto
- Establecer cronograma

## **Etapa de Análisis**

El software debe poder realizar la función principal para la que fue planeado: realizar la gestión de los gastos de manera grupal. La gestión se entiende en el caso particular como el cálculo de balances para varias personas que realizan varias transacciones como parte de un plan.

Con el fin de resaltar el valor del producto mínimo viable, se pone de manifiesto una cualidad que no había sido considerada previamente, pero se encuentra completamente cubierta: el permitir que varios usuarios tengan la posibilidad de realizar el pago inicial de una misma transacción. En las aplicaciones consultadas, tan solo una persona se hace responsable del pago y los balances se realizan de manera posterior con base en las responsabilidades establecidas para cada miembro del equipo.

Las actividades asociadas a la etapa de análisis son:

- Identificación de valor agregado del PMV

## **Etapa de diseño**

Siendo los dispositivos móviles los más utilizados por su fácil accesibilidad y portabilidad, las pantallas se diseñaron para ser visualizadas en los navegadores de dispositivos móviles. Se tiene una pantalla de Login mediante la cual el usuario ingresa a su pantalla de gestión o pantalla

principal. Desde allí puede revisar las personas que tiene inscritas para compartir gastos y los planes que tiene registrados.

Cada plan permite acceder al detalle de las transacciones que tiene asociadas, y cada transacción permite ver el detalle de los pagos y contribuciones de los participantes. También debe mostrarse el balance por cada plan, que ilustre las cantidades pendientes de cada persona.

Cada elemento (plan, persona, transacción) debe poder ser creado, editado y eliminado.

Las actividades asociadas a la etapa de diseño son:

- Diagrama entidad relación para persistencia en cliente
- Mockups de página de Login y lógica asociada
- Mockups de página de Registro de usuario y lógica asociada
- Mockups de página principal de Usuario y lógica asociada
- Mockups de página de registro de Personas y lógica asociada
- Mockups de página de registro de Planes y lógica asociada
- Mockups de página de detalle de Planes y lógica asociada
- Mockups de página de detalle de Transacciones y lógica asociada
- Mockups de página de registro de Transacciones y lógica asociada
- Mockups de página de detalle de Balance y lógica asociada

## **Etapa de Implementación**

El almacenamiento no se implementa mediante un sistema de base de datos, sino hace uso del almacenamiento en el cliente. Esto implica que cada usuario tiene datos de manera independiente, y se tendrá un solo usuario por dispositivo y podrá registrar y hacer sus cálculos de manera personal. Se utiliza el formato JSON, que se asemeja a la estructura de una base de datos no relacional, y utiliza el Local Storage del navegador de cada usuario.

Con respecto al backend, no se utiliza un lenguaje de lado de servidor. Se espera que la aplicación corra de una forma más ágil dado que se centraliza toda la lógica en el cliente, y para estos efectos se utiliza el lenguaje de programación JavaScript, de modo que el código pueda ser ejecutado por el navegador (en lugar de un servidor).

Para el frontend no se realizan mayores modificaciones. Se utiliza el lenguaje HTML5 para brindar la estructura de cada una de las vistas y la librería Bootstrap 5 en versión CDN para otorgar estilos visuales a los elementos de cada página.

La impresión de contenidos en pantalla se realiza de manera dinámica buscando hacer uso extensivo del formato JSON, brindando posibilidades de ampliación o escalamiento de la aplicación en etapas posteriores.

Las actividades asociadas a la etapa de implementación son:

- Elaboración de datos persistentes de ejemplo
- Codificación de página de Login y lógica asociada

- Codificación de página de Registro de usuario y lógica asociada
- Codificación de página principal de Usuario y lógica asociada
- Codificación de página de registro de Personas y lógica asociada
- Codificación de página de registro de Planes y lógica asociada
- Codificación de página de detalle de Planes y lógica asociada
- Codificación de página de detalle de Transacciones y lógica asociada
- Codificación de página de registro de Transacciones y lógica asociada
- Codificación de página de detalle de Balance y lógica asociada

### **Etapa de Pruebas**

Debido a los plazos, se realizan pruebas unitarias de manera manual, con una baja cobertura.

### **Etapa de Despliegue**

El despliegue de la aplicación se realiza en Github Pages.

### **Etapa de Mantenimiento**

El mantenimiento de la aplicación no se encuentra considerado en el presente proyecto.



## Diagrama de Gantt

Actividad	Inicio	Final	1-may	3-may	6-may	9-may	12-may	15-may	18-may	21-may	24-may	27-may	30-may	1-jun	3-jun	6-jun
Planeacion	1/05/2022	6/05/2022	█	█	█											
Analisis	1/05/2022	10/05/2022	█	█	█	█										
Diseño	6/05/2022	20/05/2022			█	█	█	█	█							
Implementacion	20/05/2022	29/05/2022								█	█	█				
Pruebas	29/05/2022	6/06/2022											█	█	█	█
Despliegue	2/06/2022	7/06/2022													█	█

## Tecnologías y herramientas implementadas

- **HTML:** HTML es un lenguaje que se usa para programar y desarrollar páginas webs. Sus siglas corresponden a Hyper Text Markup Language, es decir, Lenguaje de Marcas de Hipertexto, que podría ser traducido como Lenguaje de Formato de Documentos para Hipertexto. Se trata de un formato abierto que surgió a partir de las etiquetas SGML (Standard Generalized Markup Language). Concepto traducido generalmente como “Estándar de Lenguaje de Marcado Generalizado” y que se entiende como un sistema que permite ordenar y etiquetar diversos documentos dentro de una lista. HTML se encarga de desarrollar una descripción sobre los contenidos que aparecen como textos y sobre su estructura, complementando dicho texto con diversos objetos (como fotografías,

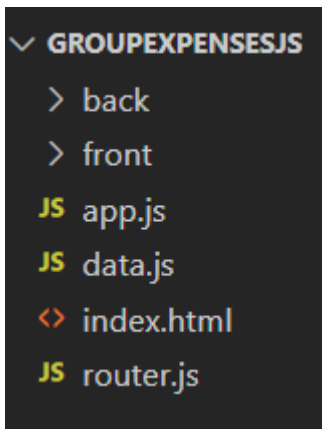
animaciones, etc.). Es un lenguaje muy simple y general que sirve para definir otros lenguajes que tienen que ver con el formato de los documentos. El texto en él se crea a partir de etiquetas, también llamadas tags, que permiten interconectar diversos conceptos y formatos. (Alvarez, 2001)

- **Bootstrap:** es un framework de desarrollo web gratuito y de código abierto. Está diseñado para facilitar el proceso de desarrollo de los sitios web responsivos y orientados a los dispositivos móviles, proporcionando una colección de sintaxis para diseños de plantillas.
- **JavaScript:** JavaScript (abreviado comúnmente "JS") es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. (Haverbeke, 2015)

## 19. SOLUCION DE INGENIERIA

Teniendo en cuenta los cambios propuestos en el plan de implementación, se construye una aplicación para gestión de gastos grupales del lado del cliente. La estructura del proyecto se muestra a continuación:

*Ilustración 12. Estructura del proyecto.*



Fuente: Elaboración propia.

- Directorio *back*: contiene la lógica temporal del backend.
- Directorio *front*: contiene la lógica del frontend.
- Archivo *app.js*: contiene las variables globales que crean instancias de los servicios y de los componentes principales, así como el llamado a la etiqueta *contentLoader*, en donde se renderiza toda la aplicación y el evento de carga de la página de log in.

- Archivo *data.js*: contiene el objeto de JavaScript con la información que simula la base de datos.
- Archivo *router.js*: contiene la lógica para enrutamiento y renderización de los componentes de la aplicación.
- Archivo *index.html*: contiene la estructura principal de la aplicación y los estilos usando el CDN de Bootstrap.

El archivo de datos tiene la siguiente estructura:

Ilustración 13. Estructura del objeto de base de datos.

```
let data = {
  user: [
    {
      id: 1,
      name: "Juan"
    }
  ],
  person: [
    {
      id: 1,
      name: "Laura"
    },
    {
      id: 2,
      name: "Juan"
    },
    {
      id: 3,
      name: "Jeison"
    }
  ],
  plan: [
    {
      id: 1,
      name: "Almuerzo despedida"
    },
    {
      id: 2,
      name: "Viaje amigos"
    }
  ]
}
```

Fuente: Elaboración propia.

El archivo de enrutador tiene la siguiente estructura que permite ejecutar los métodos comunes de todos los componentes, y renderizarlos en cualquier etiqueta destinada para tal fin:

Ilustración 14. Estructura del enrutador.

```
class Router {
  constructor() {
    this.userLogin = "/user/login"
    this.userMain = "/user/main"
    this.userMainPeople = "/user/main/people"
    this.userMainPlans = "/user/main/plans"
    this.planDetails = "/plan/details"
  }

  render(page, loader, visible){
    if(visible){
      page.createTemplate()
      loader.appendChild(page.webElements.page)
      page.createElements()
      page.readElements()
      page.writeElements()
      page.createEvents()
    }
    else
      contentLoader.removeChild(page.webElements.page)
  }

  goTo(route) {
    switch (route) {
      case this.userLogin:
        this.render(userLoginPage, contentLoader, true)
        break
      case this.userMain:
        this.render(headerPage, contentLoader, true)
        this.render(userLoginPage, contentLoader, false)
        this.render(userMainPage, contentLoader, true)
        break
      case this.planDetails:

```

Fuente: Elaboración propia.

Y el archivo de página principal tiene la siguiente estructura:

Ilustración 15. Estructura de página principal de la aplicación.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap">
    <title>Gastos compartidos</title>
  </head>
  <body>

    <div id="content-loader" class="text-center"></div>

    <script src="./data.js"></script>
    <script src="./router.js"></script>

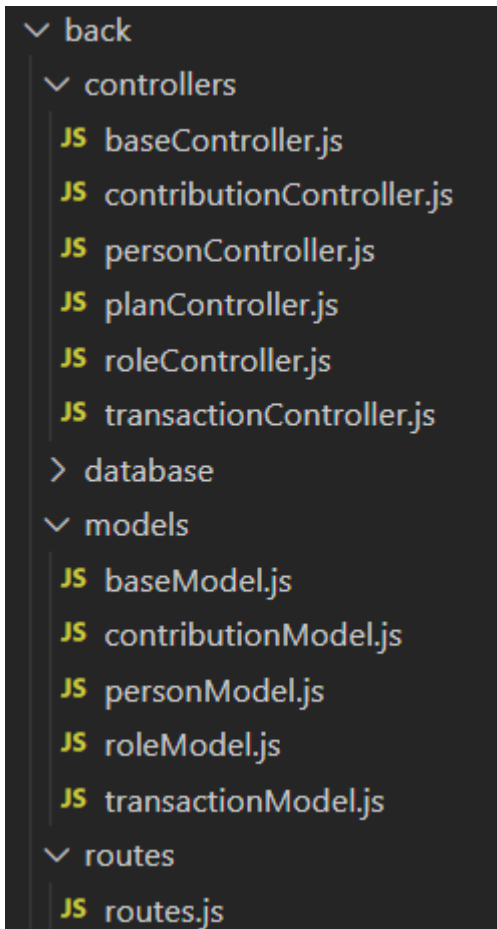
    <script src="./front/services/user-service.js"></script>
    <script src="./front/services/person-service.js"></script>
    <script src="./front/services/plan-service.js"></script>
    <script src="./front/services/transaction-service.js"></script>

    <script src="./front/user/user-login/user-login.js"></script>
    <script src="./front/common/header/header.js"></script>
    <script src="./front/person/person-card/person-card.js"></script>
  </body>
</html>
```

Fuente: Elaboración propia.

Con respecto al backend local, actualmente se encuentra desconectado y en fase de implementación. Sin embargo, se tiene la siguiente estructura:

Ilustración 16. Estructura del backend local en construcción.



Fuente: Elaboración propia.

- Directorio *controllers*: contiene todos los archivos JavaScript correspondientes a los controladores de los modelos utilizados en la aplicación, que son llamados por la lógica de rutas y ejecutan llamados para consulta de datos por medio de los modelos.

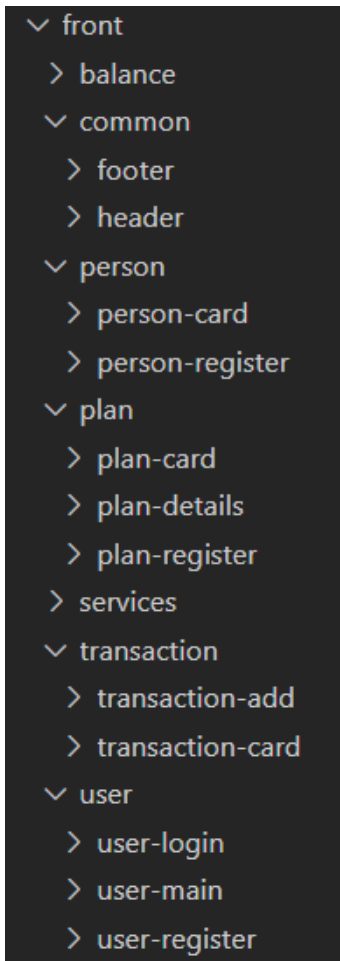


- Directorio *models*: contiene todos los archivos JavaScript correspondientes a los modelos que representan a cada una de las entidades de la base de datos. Son llamados por los controladores y sus funciones se enfocan en la realización de consultas, devolviendo objetos con la información solicitada.
- Directorio *routes*: contiene el archivo JavaScript *routes.js* que recibe peticiones de frontend y las direcciona hacia los controladores.

Con respecto al frontend, se crea una estructura basada en componentes y servicios. Los componentes se refieren a cada una de las pantallas y elementos gráficos constituyentes de la aplicación, de modo que se puedan reutilizar. Los servicios realizan los llamados a la API simulada.

El frontend tiene la siguiente estructura:

Ilustración 17. Estructura del frontend, basado en componentes.



Fuente: Elaboración propia.

Cada uno de los directorios de componentes tiene un archivo JavaScript que contiene una clase. Cada componente posee exactamente la misma estructura, que permite utilizarlo de manera individual como encadenada.

El constructor de un componente incluye una serie de objetos que permiten almacenar diferentes elementos de utilidad, a saber:

- Localizadores: definen los *id* de los elementos web que van a ser utilizados por el componente.
- Elementos Web: almacena los elementos web que serán capturados por medio de los localizadores.
- Data: almacena los datos pertinentes al componente, que serán obtenidos por medio de los servicios.
- Servicios: contiene instancias de los servicios pertinentes al componente, por medio de los cuales se obtendrá la información para renderizar.
- Enrutador: contiene la instancia del enrutador, que permite renderizar o des renderizar un componente.

Se tienen los siguientes métodos comunes:

- Crear plantilla: Contiene el código HTML para estructurar el componente. Recibe dinámicamente los identificadores para cada componente web.
- Crear Elementos: Crea los elementos web del componente, sobre los cuales se realizará algún tipo de interacción, lectura o escritura de datos.
- Leer elementos: Toma los valores ingresados por el usuario para cada elemento web y los almacena para procesamiento.
- Escribir elementos: Realiza cambios en el contenido de los elementos web con base en las consultas que se hacen por medio de los servicios.

- Crear eventos: Define que elementos web tendrán interacción con el usuario, al igual que sus comportamientos.
- Métodos personalizados: Permite crear nuevos métodos de apoyo, específicos para el componente.

Se ilustra a continuación parte del código de un componente reutilizable:

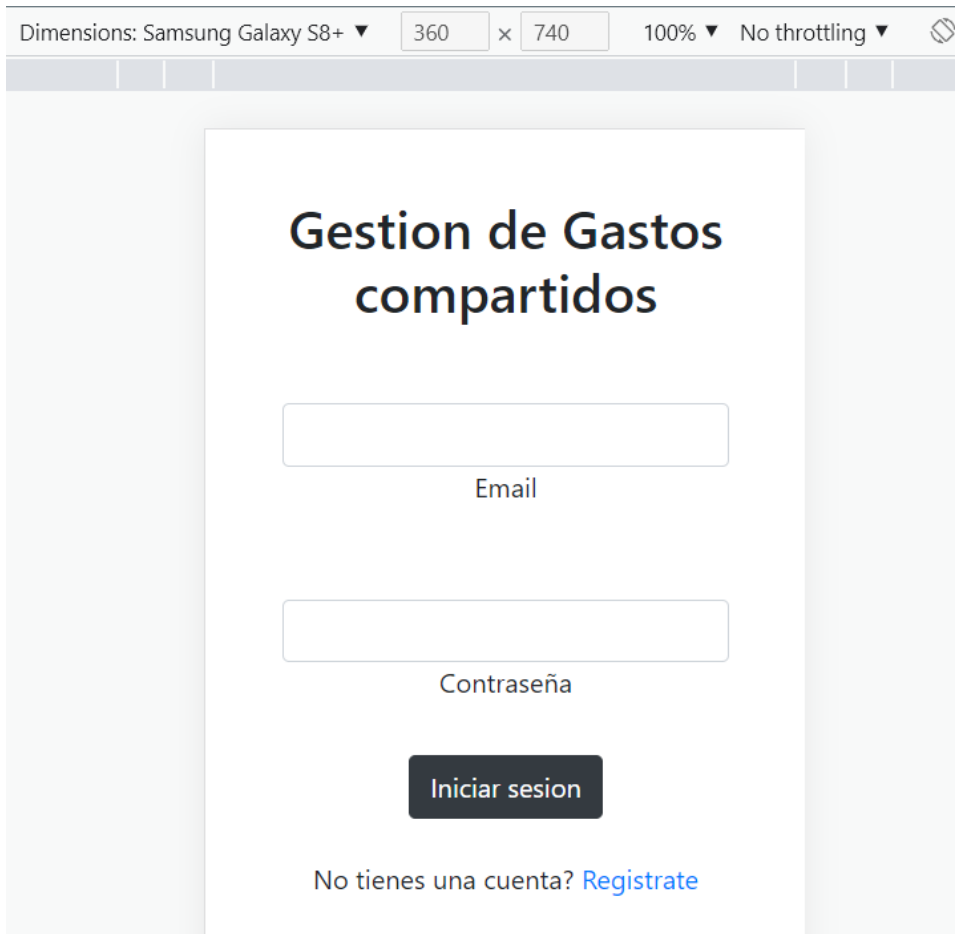
Ilustración 18. Estructura de un componente reutilizable del frontend.

```
class PlanCardPage {  
  
  constructor(plan) {  
  
    this.locators = {  
      page: `plan-card-page-${plan.id}-${plan.name}`,  
      planNameDisplay: `plan-card-plan-${plan.id}-name-display`,  
      openPlanButton: `plan-card-open-plan-${plan.id}-button`,  
    }  
  
    this.webElements = {}  
    this.data = {  
      id: plan.id,  
      name: plan.name  
    }  
  
    this.service = planService  
    this.router = router  
  
  }  
  
  createTemplate() {  
    let template = `  
      <div class="col justify-content-center text-center align-items-center">  
        <div class="row justify-content-center">  
            
        </div>  
        <div class="row justify-content-center">  
          <span id="${this.locators.planNameDisplay}" class="col-12">  
            </div>  
        </div>  
    `;  
  
    this.webElements.page = document.createElement("SECTION")  
    this.webElements.page.innerHTML = template  
    this.webElements.page.setAttribute("id", this.locators.page)  
    this.webElements.page.classList.add("m-3", "p-0")  
  }  
  
  createElements() {  
    this.webElements.planNameDisplay = document.getElementById(this.locators.planNameDisplay)  
    this.webElements.openPlanButton = document.getElementById(this.locators.openPlanButton)  
    this.webElements.editPlanButton = document.getElementById(this.locators.editPlanButton)  
    this.webElements.deletePlanButton = document.getElementById(this.locators.deletePlanButton)  
  }  
  
  readElements() {}  
  
  writeElements() {  
    this.webElements.planNameDisplay.innerHTML = this.data.name  
  }  
  
  createEvents() {  
    this.webElements.openPlanButton.addEventListener("click", () => {  
      planDetailsPage = new PlanDetailsPage(this.data)  
      this.router.goTo("/plan/details")  
    })  
  }  
}
```

Fuente: Elaboración propia.

Finalmente, se muestran algunas de las pantallas implementadas para interacción con el usuario:

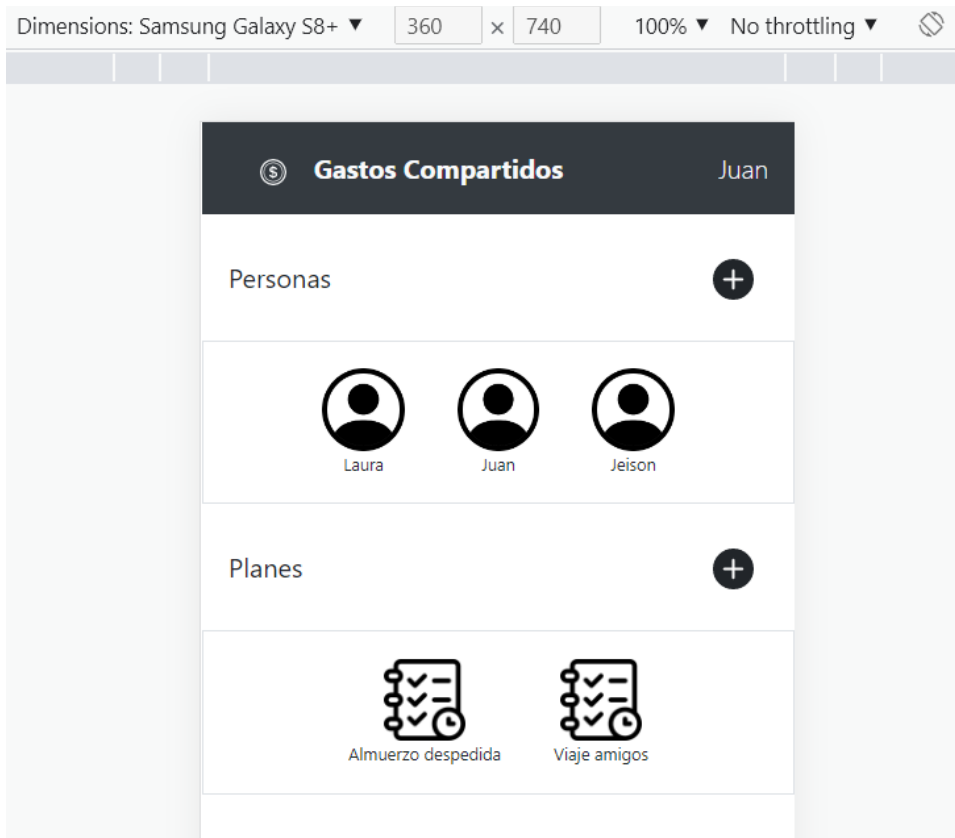
*Ilustración 19. Pantalla de inicio de sesión de la aplicación.*



The image shows a browser window displaying a login page. The browser's developer tools are open at the top, showing dimensions of 360x740 pixels at 100% zoom. The page content is centered and features the title "Gestion de Gastos compartidos" in a large, bold, black font. Below the title are two input fields: the first is labeled "Email" and the second is labeled "Contraseña". A dark gray button with the text "Iniciar sesion" is positioned below the password field. At the bottom of the form, there is a link that says "No tienes una cuenta? [Registrate](#)".

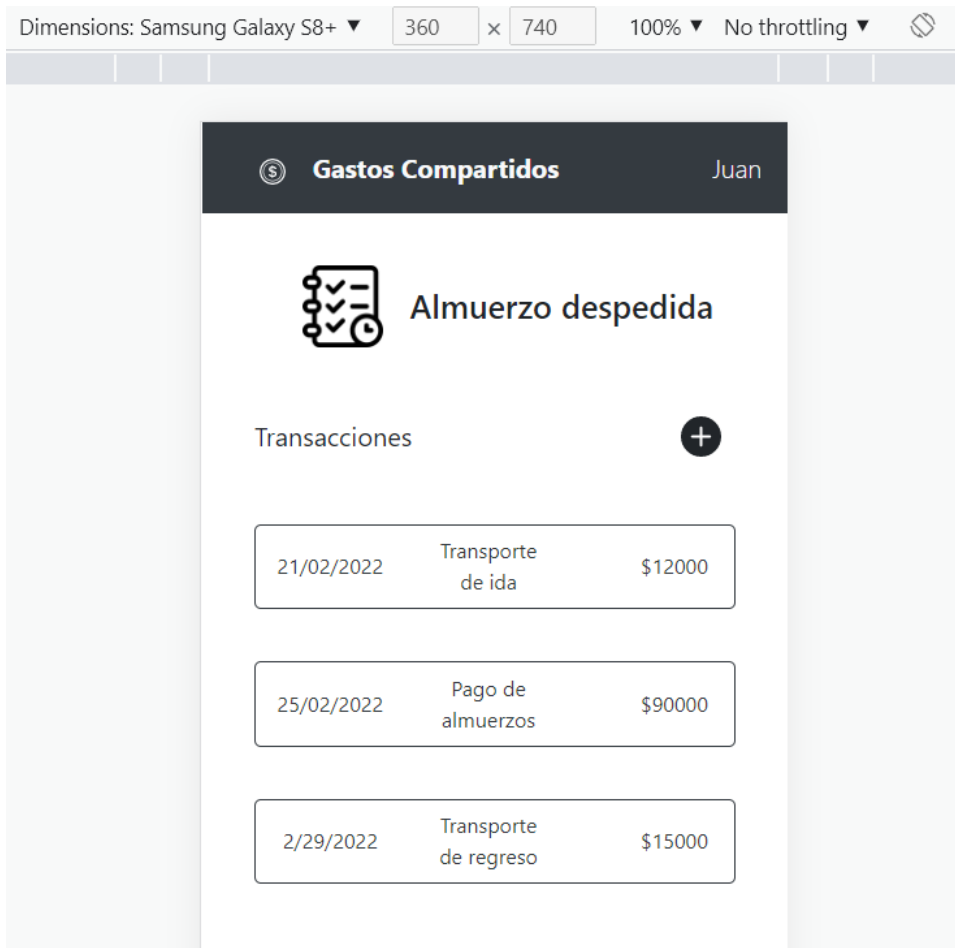
Fuente: Elaboración propia.

Ilustración 20. Pantalla principal de usuario de la aplicación.



Fuente: Elaboración propia.

Ilustración 21. Pantalla de detalles de un plan.



Fuente: Elaboración propia.



## 20. OPCIONES DE MEJORA Y TRABAJOS FUTUROS

Si bien el presente proyecto ha sido reducido, se espera que su implementación permite el escalamiento para incorporar a futuro las características que fueron omitidas, como lo son el almacenamiento en una base de datos, de ser posible en un servicio de alojamiento en la nube, y que permita la gestión participativa entre los distintos usuarios en lugar de tener un único administrador.

Los aspectos de seguridad de la aplicación pueden mejorar considerablemente, al incluir procesos más adecuados de inicio de sesión, encriptación de contraseñas y utilización de sesiones de usuario. Pueden incluirse librerías externas que faciliten el proceso.

Adicionalmente, se recomienda trasladar la lógica de la aplicación a un servidor, de modo que se aprovechen las ventajas y la modularización que un backend ofrece, junto con el debido enrutamiento y generación de endpoints para generalizar la lógica y hacer factible el uso de diversos frontend.

Se recomienda realizar un análisis de riesgo previo al inicio del proyecto y definir de igual forma las medidas de control y contingencia para evitar fuertes impactos en el desempeño, alcances y modificación de cronogramas.

Finalmente, se enfatiza en la realización de pruebas de manera adecuada desde el comienzo del ciclo de vida del proyecto, para asegurar un producto de calidad.

## 21. CONCLUSIONES

- En este proyecto se desarrolla una aplicación para analizar, controlar, registrar, prever y calcular cualquier tipo de gasto que ocurra por cualquier usuario dentro de un grupo de individuos. Todo esto con la finalidad de evitar posibles pérdidas económicas y malentendidos con las personas que deseemos pagar alguna suma monetaria.
- El diseño de la aplicación se realiza de manera simple y amigable, haciéndola intuitiva y de fácil uso, adicionalmente que es una aplicación enfocada en los gastos grupales.
- Durante el desarrollo del proyecto se ponen en práctica los conocimientos adquiridos en torno a la estructura de datos y algoritmos, programación, competencias de programación y desarrollo centrado en los usuarios, para obtener una ampliación que satisfaga los estándares esperados y propuestos.
- Se encuentran alternativas para solucionar el problema dados cambios relevantes en la estructura del equipo de desarrollo del proyecto.
- Se identifica una fuente de valor agregado del proyecto en la posibilidad de hacer registros por parte de varias personas para el pago inicial de un gasto.
- Mediante la estrategia de selección de alternativas se define que una aplicación web era la plataforma más apropiada para cumplir con los objetivos del presente trabajo, principalmente por temas de costos y cobertura de dispositivos, a comparación de una aplicación móvil.

## 22. LISTA DE REFERENCIAS

Álvarez, Miguel (2020) Que es MVC. Desarrolloweb.com.

<https://desarrolloweb.com/articulos/que-es-mvc.html>

Álvarez, Miguel (2020) Que es HTML. Desarrolloweb.com. Tomado de

<https://desarrolloweb.com/articulos/que-es-html.html>

Billr (s.f.) Aplicación móvil (iPhone) para división de gastos en bares y restaurantes. Sitio web:

<http://billr.me/> Consultado el 19 de febrero de 2022.

Pantaleo, G., & Rinaudo, L. (2015). Ingeniería de software. Alfaomega Grupo Editor.

BillZer (s.f.) Aplicación web para control de gastos. <https://billzer.com/> Consultado el 19 de febrero de 2022.

BlueCoins (s.f.) Finance & Budget. Aplicación para seguimiento y control de gastos personales.

<https://www.bluecoinsapp.com/guide/> Consultado el 19 de febrero de 2022.

Chaudhry, Vishal (2022) *Joint Expense. Record & Report Group Expenses*. Aplicación para

iPad. <https://apps.apple.com/us/app/luna-split-bills-expenses/id1535754708>

Cloud Height Technologies Limited (2022) *Group Expense – track & Split*. Aplicación para

dividir y registrar cuentas (Mac App Store) <https://apps.apple.com/us/app/group-expense-track-split/id1150902475>

Coa, Richard, (2013) Modelos Cliente – Servidor. Ministerio del poder popular para la educación universitaria

Cormen Thomas H.; Leiserson Charles E.; Rivest Ronald L.; Stein Clifford (2001) *Introduction to Algorithms*. Chapter 35.5: The subset-sum problem. MIT Press, 2nd edition.

Haverbeke, M. (2015). Eloquent JavaScript: A Modern Introduction to Programming. Obtenido de <https://eloquentjavascript.net/>

KittySplit (2022) Aplicación web para llevar el control de los gastos compartidos.  
<https://www.kittysplit.com/es/>

Kocken, Stan (2017) *Sesterce – Split expenses. For a trip or everyday life*. Aplicacion para iPad. App Store. <https://apps.apple.com/us/app/sesterce-split-expenses/id1239566667>

Koman, Milan. Ticha, Marie. (1998) Travelling Together and Sharing Expenses (Examples of Investigation of Situations). *Teaching Mathematics and Its Applications: International Journal of the IMA*. pp117 – 122.

Kuzovkov, Alexey (2022) *2Split. Group Expenses*. App Store. Aplicacion para iPhone.  
<https://apps.apple.com/us/app/2split/id1234044508>

Kuzyk, Kyrylo (2022) *Split Together. Split bills and share expenses*. Aplicacion para iPhone.  
<https://apps.apple.com/us/app/spend-together/id1446549608>

Luna App Inc. (2020) *Luna: Split Bills & Expenses. Tracker, splitter, organizer*. Application for iPhone. <https://apps.apple.com/us/app/luna-split-bills-expenses/id1535754708>

Marques, Luis (2017) *Share Expenses with Friends*. App Store. Aplicacion para iPad. <https://apps.apple.com/us/app/share-expenses-with-friends/id1231171881>

Parra, Javier., Garrigues, Carles., Martin, Eduard (2019) *AllPay: aplicacion para gastos compartidos en Android*. Universitat Oberta de Catalunya (UOC).

Roque, Dianelys Del Valle (2022) Estimación de costos de desarrollo de software. Recurso online, disponible en: <https://www.gestiopolis.com/estimacion-de-costos-de-desarrollo-de-software>, consultado el 29 de mayo de 2022.

Settle Up (s.f.) Aplicación movil para gastos grupales. (Play Store) [https://play.google.com/store/apps/details?id=cz.destil.settleup&hl=en\\_US&gl=US](https://play.google.com/store/apps/details?id=cz.destil.settleup&hl=en_US&gl=US)  
Consultado el 19 de febrero de 2022.

Split (s.f.) Administrador de gastos grupales (Mac App Store) <https://apps.apple.com/es/app/split-group-expenses-manager/id806813572> Consultado el 19 de febrero de 2022.

SplitWise (s.f.) Aplicación movil para control de gastos grupales (Play Store) [https://play.google.com/store/apps/details?id=com.Splitwise.SplitwiseMobile&hl=en\\_US&gl=US](https://play.google.com/store/apps/details?id=com.Splitwise.SplitwiseMobile&hl=en_US&gl=US) Consultado el 19 de febrero de 2022.

Steffgriffy (s.f.) Split Group Bills - Manage Expenses - Share Costs.

<https://www.amazon.es/Split-Group-Bills-Manage-Expenses/dp/B0842VRK47>

Consultado el 19 de febrero de 2022.

Tricount (2022) Mobile and web application. Make sharing expenses simple for travel, activities, and daily life. <https://www.tricount.com/en/organizing-group-expenses-among-friends>

Consultado el 19 de febrero de 2022.

Tsybulenka, Natalia (2020) *Splity – Group Expenses. Split bills and costs*. Aplicación para iPhone. <https://apps.apple.com/us/app/splity-group-expenses/id1512670632>

Vávra, David (2012) *Mobile Application for Group Expenses and Its Deployment*. Master's thesis. Czech Technical University in Prague. Faculty of Electrical Engineering. Department of Computer Graphics and Interaction.

Villar, Pablo (2022). *Vaquitas. Split group expenses quickly*. Aplicacion para iPad. App Store. <https://apps.apple.com/us/app/vaquitas/id1406144934>

Ye-jin Jang, Yu-min Jo, Ji-in Shin, Yun-jeong Jang, Da-woon Jeong, & Jong-ho Paik. (2021). Design and Implementation of Disinfection Service Platform based on MVC Pattern Using Web/App. *Journal of Korean Society for Internet Information*, 22(6), 41–49. <https://doi-org.bdbiblioteca.universidadean.edu.co/10.7472/jksii.2021.22.6.41>



## ANEXO ENCUESTA

¿Has sufrido algún tipo de malentendido con tus amigos, familiares o pareja al momento de dividir los gastos en los viajes o salidas comunes?

¿Qué tipo de malentendido has sufrido por dividir los gastos o por no controlar tus gastos?

¿Conoces alguna aplicación o página web que se dedique a dividir los gastos en las salidas de viajes o cotidianas con tus amigos o familiares?

¿Qué mecanismos o herramientas tecnológicas usas al momento de compartir los gastos?

¿Llevas un registro de los gastos asumidos de tu parte y tus amigos en los viajes o salidas comunes?

¿Te gustaría crear un bote común virtual desde el que se puedan hacer pagos?

¿Pagarías más de \$20.000 mensuales por servicios premium como exportación de archivos de tus cuentas de gastos compartidos, evitar anuncios y una mejor experiencia en aplicaciones de gastos compartidos?