

Documento de Ejecución

“FamCash”

Responsables: Díaz Alvizuri Luis Fabian,
Huarcaya Lizarraga Astrid,
Pinto Gálvez Sofía Alejandra,
Retamozo Vasquez Santiago Sebastian,
Rosas Lipa Gianella Ariana,
Zeballos Huayna Diego Alonso.

Versión	Fecha	Descripción	Elaboradores
1.0 Descargar Archivo de GitHub	18/09/2025	Primera versión	Huarcaya Lizarraga Astrid Judith Pinto Gálvez Sofía Alejandra Zeballos Huayna Diego Alonso
2.0 Descargar Archivo de GitHub	11/10/2025	Segunda versión	Pinto Gálvez Sofía Alejandra Díaz Alvizuri Luis Fabian Rosas Lipa Gianella Ariana Retamozo Vasquez Santiago Sebastian
3.0	09/11/2025	Tercera versión	Rosas Lipa Gianella Ariana Retamozo Vasquez Santiago Sebastian Pinto Gálvez Sofía Alejandra

Cliente:

- Guillermo Enrique Calderón Ruiz

Índice General

1. Introducción.....	4
2. Sección Administrativa.....	4
2.1 Codificación.....	4
2.1.1 Requisitos.....	4
2.1.2 Casos de Uso.....	5
2.1.3 Objetos.....	5
2.1.4 Interfaces.....	6
2.1.5 Gestores.....	6
2.1.6 Entidades.....	7
2.1.7 Diagrama de Clases de Análisis.....	7
2.1.8 Diagramas de Secuencia.....	8
2.1.9 Diagrama de Clases de Diseño.....	8
2.1.10 Modelo de Datos.....	9
2.2 Matriz de Rastreo de Requisitos.....	9
3. Análisis.....	10
3.1 Requisitos de Usuario.....	10
3.1.1 Requisitos Funcionales.....	10
3.1.2 Requisitos No Funcionales.....	12
3.2 Requisitos de Sistema.....	16
3.2.1 Requisitos Funcionales.....	16
3.2.2 Requisitos No Funcionales.....	19
3.3 Modelo de Requisitos.....	21
3.3.1 Modelo de Comportamiento.....	21
3.3.1.a Diagrama de Casos de Uso y Especificación.....	21
3.3.2 Modelo de Presentación.....	23
3.3.2.a Diagrama de Navegabilidad.....	23
3.3.3 Modelo de Información.....	24
3.3.3.a Diagrama de Clases de Análisis.....	24
4. Diseño.....	25
4.1 Diagramas de Secuencia.....	25
4.2 Diagrama de Clases de Diseño.....	26
4.3 Modelo Entidad-Relación.....	27
4.3.1 Diagrama del Modelo Entidad-Relación.....	27
4.4 Modelo Relacional.....	28
4.4.1 Diagrama del Modelo Relacional.....	28
4.5 Plan de Pruebas.....	29
5. Construcción.....	30
5.1 Modelo Físico.....	30
5.1.1 Diagrama del Modelo Físico.....	30
5.1.2 Proyección de Crecimiento.....	31
5.1.3 Copias de Seguridad.....	32
5.1.4 Implementación del Modelo Físico.....	33
5.2 Producto.....	34

6. Referencias.....	34
----------------------------	-----------

1. Introducción

Este es el documento de ejecución para el desarrollo del sistema de administración de una economía familiar FamCash, propiedad de la empresa SOLVE-IT. Está estructurado en seis etapas principales: Análisis, Diseño, Construcción, Testing, Despliegue y Mantenimiento, las cuales conforman el ciclo de vida del software. Cada etapa responde a un orden lógico y secuencial que permite organizar el desarrollo del sistema de forma clara, orientada al cumplimiento de los objetivos planteados por la empresa.

2. Sección Administrativa

Esta sección aborda los aspectos administrativos que permiten una gestión integral y ordenada de todos los elementos de FamCash.

2.1 Codificación

A continuación se especifican las directrices para la asignación de nombres y códigos a los distintos componentes del sistema FamCash.

2.1.1 Requisitos

Los requisitos son codificados con la letra R seguida de la primera letra de las palabras si es funcional o no funcional y se les asigna un número único y secuencial.

Tabla 1.1 Codificación de requisitos

Código	Nombre
RF-01	Registrar Entrada Diaria
RF-02	Ver Balance
RF-03	Gestión de conceptos
RF-04	Gestión de perfil
RF-05	Ingreso al aplicativo
RF-06	Gestión de perfiles familiares
RF-07	Salida del aplicativo

2.1.2 Casos de Uso

Los casos de uso son codificados con el nombre CU y se les asigna un número único y secuencial.

Tabla 1.2 Codificación de casos de uso

Código	Nombre
CU-001	Registrarse
CU-002	Iniciar Sesión
CU-003	Selección de Perfil
CU-004	Configurar Perfil
CU-005	Cerrar Sesión
CU-006	Gestionar Perfiles Familiares
CU-007	Crear Perfil
CU-008	Editar Perfil
CU-009	Eliminar Perfil
CU-010	Gestionar Conceptos
CU-011	Crear Concepto
CU-012	Editar Concepto
CU-013	Deshabilitar Concepto
CU-014	Registrar Entrada Diaria
CU-015	Ver Balance Usuario
CU-016	Ver Balance Administrador

2.1.3 Objetos

Los objetos son codificados con el nombre OBJ y se les asigna un número único y secuencial.

Tabla 1.3 Codificación de objetos

Código	Nombre
OBJ-01	Usuario
OBJ-02	Administrador

2.1.4 Interfaces

Las interfaces son codificadas con el nombre UI y se le asigna un número único y secuencial.

Tabla 1.4 Codificación de interfaces

Código	Nombre
UI-01	UIPáginaDeInformación
UI-02	UIRegistro
UI-03	UIIniciarSesión
UI-04	UISelecciónDePerfiles
UI-05	UIVistaEntradaDiaria
UI-06	UIVistaBalance
UI-07	UIConfiguraciónConceptosCrear
UI-08	UIConfiguraciónConceptosEditar
UI-09	UIConfiguraciónDePerfil
UI-10	UIConfiguraciónDePerfilesFamiliaresAdministradorCrear
UI-11	UIConfiguraciónDePerfilesFamiliaresAdministradorEditar

2.1.5 Gestores

Los gestores son codificados con el nombre GES y se les asigna un número único y secuencial.

Tabla 1.5 Codificación de gestores

Código	Nombre
GES-001	GESGestorUsuarios
GES-002	GESGestorPerfiles
GES-003	GESGestorSesiones
GES-004	GESGestorConceptos
GES-005	GESGestorTransacciones

2.1.6 Entidades

Las entidades en la base de datos son codificadas con el nombre TAB y se les asigna un número único y secuencial.

Tabla 1.6 Codificación de entidades

Código	Nombre
TAB-001	TABUsuarios
TAB-002	TABPerfiles
TAB-003	TABConceptos
TAB-004	TABTransacciones

2.1.7 Diagrama de Clases de Análisis

Los diagramas de comunicación son codificados con el nombre DCA y se les asigna un número único y secuencial.

Tabla 1.7 Codificación de clases de análisis

Código	Nombre
DCA-001	Registrarse
DCA-002	Iniciar Sesión
DCA-003	Selección de Perfil
DCA-004	Configurar Perfil
DCA-005	Cerrar Sesión
DCA-006	Crear Perfil
DCA-007	Editar Perfil
DCA-008	Eliminar Perfil
DCA-009	Crear Concepto
DCA-010	Editar Concepto
DCA-011	Deshabilitar Concepto
DCA-012	Registrar Entrada Diaria
DCA-013	Ver Balance Usuario
DCA-014	Ver Balance Administrador

2.1.8 Diagramas de Secuencia

Los diagramas de secuencia son codificados con el nombre SD y se les asigna un número único y secuencial.

Tabla 1.8 Codificación de diagramas de secuencia

Código	Nombre
SD-01	Registrarse
SD-02	Iniciar Sesión
SD-03	Selección de Perfiles
SD-04	Configurar Perfil
SD-05	Cerrar Sesión
SD-06	Crear Perfil
SD-07	Editar Perfil
SD-08	Eliminar Perfil
SD-09	Crear Concepto
SD-10	Editar Concepto
SD-11	Deshabilitar Concepto
SD-12	Registrar Entrada Diaria
SD-13	Ver Balance Usuario
SD-14	Ver Balance Administrador

2.1.9 Diagrama de Clases de Diseño

Los diagramas de clases de diseño son codificados con el nombre DC y se les asigna un número único y secuencial.

Tabla 1.9 Codificación de clases de diseño

Código	Nombre
DC-01	Diagrama de Clases FamCash

2.1.10 Modelo de Datos

Tabla 1.10 Codificación de modelo de datos

Código	Nombre
MER-001	Modelo entidad-relación
MR-001	Modelo relacional
MF-001	Modelo físico

2.2 Matriz de Rastreo de Requisitos

La siguiente matriz de rastreo de requisitos permite establecer una correspondencia clara entre los requisitos definidos y cada artefacto realizado durante el tiempo de trabajo. La Figura 1 muestra una vista previa de la matriz de rastreo de requisitos de FamCash.

Figura 1 Matriz de rastreo de requisitos

Código de requisitos	Requisitos	Descripción	Artefactos									
			Diagrama de Entidad-Relación	Diagrama de Relación	Diagrama de Normalización	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos
MR-001	Requisito de Entidad-Relación	El sistema permite registrar, actualizar y eliminar información de los usuarios (nombre, correo electrónico, contraseña, etc.) y de los productos (nombre, precio, etc.)	Diagrama de Entidad-Relación	Diagrama de Relación	Diagrama de Normalización	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos
MR-002	Requisito de Relación	El sistema permite consultar los datos de los usuarios y de los productos, así como la relación entre ellos (usuarios que han comprado productos).	Diagrama de Entidad-Relación	Diagrama de Relación	Diagrama de Normalización	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos
MR-003	Requisito de Base de Datos	El sistema permite almacenar la información de los usuarios y de los productos en una base de datos relacional.	Diagrama de Entidad-Relación	Diagrama de Relación	Diagrama de Normalización	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos
MR-004	Requisito de Base de Datos	El sistema permite consultar la información de los usuarios y de los productos en una base de datos relacional.	Diagrama de Entidad-Relación	Diagrama de Relación	Diagrama de Normalización	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos
MR-005	Requisito de Base de Datos	El sistema permite almacenar la información de los usuarios y de los productos en una base de datos relacional.	Diagrama de Entidad-Relación	Diagrama de Relación	Diagrama de Normalización	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos
MR-006	Requisito de Base de Datos	El sistema permite consultar la información de los usuarios y de los productos en una base de datos relacional.	Diagrama de Entidad-Relación	Diagrama de Relación	Diagrama de Normalización	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos
MR-007	Requisito de Base de Datos	El sistema permite almacenar la información de los usuarios y de los productos en una base de datos relacional.	Diagrama de Entidad-Relación	Diagrama de Relación	Diagrama de Normalización	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos
MR-008	Requisito de Base de Datos	El sistema permite consultar la información de los usuarios y de los productos en una base de datos relacional.	Diagrama de Entidad-Relación	Diagrama de Relación	Diagrama de Normalización	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos
MR-009	Requisito de Base de Datos	El sistema permite almacenar la información de los usuarios y de los productos en una base de datos relacional.	Diagrama de Entidad-Relación	Diagrama de Relación	Diagrama de Normalización	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos
MR-010	Requisito de Base de Datos	El sistema permite consultar la información de los usuarios y de los productos en una base de datos relacional.	Diagrama de Entidad-Relación	Diagrama de Relación	Diagrama de Normalización	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos	Diagrama de Base de Datos

Link de matriz de rastreo de requisitos: [Abrir matriz de rastreo de requisitos](#)

3. Análisis

En esta sección se detalla la fase de análisis de FamCash, abarcando tanto los requisitos funcionales como los no funcionales, así como el modelo de análisis. Este último comprende los modelos de comportamiento, presentación e información, proporcionando así una visión integral de la estructura y funcionamiento del sistema.

3.1 Requisitos de Usuario

Los requisitos de usuario representan las necesidades, expectativas y objetivos expresados por los usuarios finales en un lenguaje claro y comprensible. Reflejan lo que esperan obtener del sistema en cuanto a funciones o servicios.

3.1.1 Requisitos Funcionales

Las siguientes tablas muestran los requisitos funcionales del usuario de FamCash.

Tabla 2.1 Requisitos Funcionales del Usuario

Código	Nombre	Descripción	Prioridad
RU-01-Registro de Usuarios	Registro de Usuarios	Una persona se registra como nuevo cliente, proporcionando nombre completo, contraseña segura y nuevamente repetir la contraseña. El sistema debe verificar que el username no esté registrado previamente.	Alta
RU-02-Inicio de Sesión	Inicio de Sesión	Un usuario debe poder iniciar sesión con su nombre y contraseña previamente registrados.	Alta
RU-03-Selección de Perfil Familiar	Selección de Perfil Familiar	Una vez autenticado, el usuario puede seleccionar entre los diferentes perfiles familiares que han sido registrados y a los que tiene acceso. Esta selección determina el contexto de operación (usuario estándar o administrador) y los datos financieros que podrá visualizar y gestionar. Al tratar de ingresar a uno de estos te pide contraseña si es que la tiene.	Alta
RU-04-Registrar Ingreso Diario	Registrar Ingreso Diario	El usuario debe poder registrar un ingreso especificando monto (número positivo), concepto (seleccionable de lista o desarrollando un concepto rápido), fecha y descripción opcional.	Alta
RU-05-Registrar Egreso Diario	Registrar Egreso Diario	El usuario debe poder registrar un gasto especificando monto (número positivo), concepto (seleccionable de lista o desarrollando un concepto rápido), fecha y descripción opcional. El sistema debe validar que el monto sea numérico.	Alta

Tabla 2.2 Requisitos Funcionales del Usuario

Código	Nombre	Descripción	Prioridad
RU-06-Guardar Transacciones	Guardar Transacciones	El usuario debe poder guardar permanentemente las transacciones registradas mediante un botón de guardado, con confirmación visual del éxito de la operación.	Alta
RU-07-Ver Balance Personal	Ver Balance Personal	El usuario debe poder visualizar su balance financiero individual mostrando total de ingresos, total de egresos, balance neto y lista de transacciones recientes con filtros por período.	Alta
RU-08-Ver Balance Familiar	Ver Balance Familiar	El administrador debe poder visualizar el balance consolidado de todos los miembros de la familia, incluyendo totales por persona y balance general familiar.	Alta
RU-09-Filtrar por Fechas	Filtrar por Fechas	El usuario debe poder filtrar las consultas de balance seleccionando fecha fin mediante calendarios desplegables para ver movimientos en un período específico.	Alta
RU-10-Crear Conceptos	Crear Conceptos	El usuario debe poder crear nuevos conceptos especificando nombre único, tipo (ingreso/egreso), categoría y periodicidad. El sistema debe validar que no exista un concepto con el mismo nombre.	Alta
RU-11-Editar Conceptos	Editar Conceptos	El usuario debe poder modificar conceptos existentes cambiando nombre, tipo, categoría o periodicidad. El sistema debe mostrar los datos actuales y permitir su edición.	Alta
RU-12-Deshabilitar Concepto	Deshabilitar Concepto	El usuario debe poder deshabilitar conceptos existentes con confirmación previa. Ya no aparecerá el concepto en ningún nuevo registro diario sólo en transacciones previas a su deshabilitación.	Media
RU-13-Habilitar Concepto	Habilitar Concepto	El usuario debe poder habilitar conceptos deshabilitados previamente con confirmación previa. El concepto aparecerá de manera normal en los registros diarios nuevamente dependiendo de su periodo.	Media

Tabla 2.3 Requisitos Funcionales del Usuario

Código	Nombre	Descripción	Prioridad
RU-14-Configurar Perfil Personal	Configurar Perfil Personal	El usuario debe poder actualizar su información personal incluyendo nombre y contraseña, con validación de contraseña actual antes de permitir cambios.	Media
RU-15-Crear Perfil Familiar	Crear Perfil Familiar	El administrador debe poder registrar nuevos miembros familiares especificando nombre, contraseña inicial y nivel de permisos (usuario/administrador).	Media
RU-16-Editar Perfil Familiar	Editar Perfil Familiar	El administrador debe poder otorgar permisos a diferentes miembros de la familia. Quitar permisos o dar permisos de administrador	Media
RU-17-Eliminar Perfil Familiar	Eliminar Perfil Familiar	El administrador debe poder eliminar perfiles de miembros familiares con confirmación doble. El sistema debe archivar las transacciones del perfil eliminado para mantener el historial.	Media
RU-18-Cerrar sesión Perfil Personal	Cerrar Sesión	El usuario debe poder salir del sistema de forma segura mediante un botón de cerrar sesión que termine su sesión activa y redirige a la parte de selección de perfiles.	Alta
RU-19-Cerrar sesión de Cuenta familiar	Cerrar sesión de perfil familiar.	El usuario debe poder salir de su perfil familiar mediante un botón de cerrar, que termina su sesión	Alta

3.1.2 Requisitos No Funcionales

Las siguientes tablas muestran los requisitos no funcionales del usuario de FamCash.

Tabla 3.1 Requisitos No Funcionales del Usuario

Código	Nombre	Descripción	Prioridad
RUN-01-Facilidad de uso	Facilidad de uso	El sistema debe proporcionar una experiencia de usuario sencilla e intuitiva, permitiendo que personas sin conocimientos previos de computación puedan realizar todas las operaciones básicas de registro de ingresos, egresos y consulta de balances sin necesidad de capacitación técnica o manuales complejos. La interfaz debe guiar naturalmente al usuario a través de los procesos principales del sistema.	Alta

Tabla 3.2 Requisitos No Funcionales del Usuario

Código	Nombre	Descripción	Prioridad
RUN-02-Navegación	Navegación	La estructura de navegación debe seguir patrones familiares y lógicos, con menús claros, breadcrumbs visibles y flujos de trabajo coherentes entre pantallas. Los usuarios deben poder moverse entre las diferentes secciones (entrada diaria, balance, configuración) sin confusión y siempre saber dónde se encuentran dentro del sistema y cómo regresar a pantallas anteriores.	Alta
RUN-03-Rapidez de registro	Rapidez de registro	Los formularios para el registro de movimientos financieros diarios (ingresos y egresos) deben estar optimizados para completarse en un tiempo máximo de 2 minutos, incluyendo la selección de conceptos, ingreso de montos y confirmación. El sistema debe minimizar el número de clicks y campos obligatorios, ofreciendo valores predeterminados inteligentes y recordando preferencias del usuario.	Media
RUN-04-Iconografía comprensible	Iconografía comprensible	Todos los iconos, símbolos y elementos gráficos de la interfaz deben ser universalmente reconocibles y autodescriptivos, evitando la necesidad de explicaciones adicionales o tooltips. Se debe utilizar iconografía estándar para operaciones comunes (guardar, editar, eliminar, configurar).	Media
RUN-05-Respuesta rápida	Respuesta rápida	El tiempo de respuesta del sistema para todas las operaciones de consulta de información financiera (balances, historiales, reportes) debe ser imperceptible para el usuario, proporcionando feedback inmediato a cada acción. Las consultas complejas no deben exceder los 3 segundos, y las operaciones básicas deben completarse en menos de 1 segundo para mantener la fluidez de la experiencia de usuario.	Alta

Tabla 3.3 Requisitos No Funcionales del Usuario

Código	Nombre	Descripción	Prioridad
RUN-06-Carga instantánea	Carga instantánea	La visualización de datos financieros almacenados (movimientos diarios, balances mensuales, configuraciones de perfiles) debe aparecer de forma inmediata al usuario, sin pantallas de carga visibles o tiempos de espera prolongados.	Alta
RUN-07-Visualización fluida	Visualización fluida	Las transiciones entre pantallas, la carga de nuevos contenidos y los cambios de estado en la interfaz deben ocurrir de manera suave y sin interrupciones visibles. No debe haber parpadeos, saltos de contenido o retrasos perceptibles que interrumpen el flujo natural de trabajo del usuario durante la gestión de su economía familiar.	Media
RUN-08-Privacidad de datos	Privacidad de datos	Toda la información financiera personal y familiar debe mantenerse estrictamente confidencial y protegida contra accesos no autorizados. El sistema debe garantizar que terceros no puedan acceder, visualizar o manipular la información financiera almacenada.	Alta
RUN-09-Control de acceso	Control de acceso	El perfil de administrador familiar debe tener la capacidad exclusiva de visualizar información financiera consolidada de todos los miembros de la familia, incluyendo movimientos individuales, balances personales y reportes familiares completos. Esta funcionalidad debe estar claramente diferenciada y protegida, asegurando que solo el administrador designado pueda acceder a esta vista integral del estado financiero familiar.	Alta
RUN-10-Acceso personal	Acceso personal	Cada miembro de la familia debe tener acceso únicamente a su información financiera personal (sus propios ingresos, egresos, balance individual) sin poder visualizar los datos económicos de otros miembros familiares. El sistema debe implementar una separación clara de datos por usuario, asegurando la privacidad individual dentro del entorno familiar compartido.	Alta

Tabla 3.4 Requisitos No Funcionales del Usuario

Código	Nombre	Descripción	Prioridad
RUN-11-Permanencia de datos	Permanencia de datos	El sistema debe garantizar la integridad y permanencia de todos los datos financieros ingresados por los usuarios, incluyendo movimientos diarios, configuraciones de conceptos y perfiles familiares. No debe existir pérdida de información por fallos del sistema, cierres inesperados o errores de usuario. Todos los datos deben estar respaldados automáticamente y ser recuperables en cualquier momento.	Alta
RUN-12-Funcionamiento estable	Funcionamiento estable	El sistema debe operar de manera consistente y confiable durante el uso diario continuo, sin presentar errores, fallos o comportamientos inesperados que interrumpan las actividades de gestión financiera familiar. Debe mantener un rendimiento uniforme independientemente de la cantidad de datos almacenados o el número de operaciones realizadas por los usuarios.	Alta
RUN-13-Precisión de cálculos	Precisión de cálculos	Todos los cálculos financieros realizados por el sistema (sumas de ingresos, egresos, balances, totales mensuales y anuales) deben ser matemáticamente exactos y confiables. No debe existir margen de error en las operaciones aritméticas, y los resultados deben ser consistentes en todas las vistas y reportes del sistema, garantizando la confianza de los usuarios en la información presentada.	Alta
RUN-14-Disponibilidad continua	Disponibilidad continua	El sistema debe estar accesible y operativo en los momentos que la familia necesite registrar movimientos financieros o consultar información económica, especialmente durante horarios habituales de uso doméstico. Debe minimizar periodos de inactividad o mantenimiento que puedan interferir con la rutina diaria de gestión financiera familiar.	Media

Tabla 3.5 Requisitos No Funcionales del Usuario

Código	Nombre	Descripción	Prioridad
RUN-15-Interfaz legible	Interfaz legible	Todos los elementos visuales del sistema (textos, números, botones, menús) deben tener un tamaño, contraste y tipografía adecuados para garantizar la legibilidad clara en diferentes condiciones de uso. Los elementos interactivos deben ser lo suficientemente grandes para ser activados fácilmente, y la información debe estar organizada de manera que facilite la lectura y comprensión rápida de los datos financieros presentados.	Media

3.2 Requisitos de Sistema

En esta sección se detallan los requisitos de sistema que guiarán el desarrollo del proyecto, especificando las condiciones técnicas y operativas necesarias para su implementación exitosa.

3.2.1 Requisitos Funcionales

Las siguientes tablas muestran los requisitos funcionales del sistema FamCash.

Tabla 4.1 Requisitos Funcionales del Sistema

Código	Nombre	Descripción	Prioridad
RF-01	Registrar Entrada Diaria	El sistema debe permitir al usuario registrar sus movimientos financieros diarios (ingresos y egresos) mediante una interfaz que presenta los conceptos habilitados para ingresar montos. En esta misma interfaz el sistema debe permitir al usuario consultar y editar las entradas registradas en días anteriores mediante la selección de una fecha diferente en el calendario. Adicionalmente, se incluye el proceso de guardado, que recopila los datos ingresados y los almacena como transacciones asociadas al perfil y la fecha seleccionada.	Alta

Tabla 4.2 Requisitos Funcionales del Sistema

Código	Nombre	Descripción	Prioridad
RF-02	Ver Balance	<p>Consulta de balance individual: El sistema debe permitir al usuario visualizar su situación financiera personal mediante una interfaz que muestre el historial de ingresos y egresos registrados. Adicionalmente el sistema debe permitir al usuario seleccionar una fecha específica a través de un calendario interactivo, actualizando la vista para mostrar las transacciones y el balance correspondiente a dicha fecha.</p> <p>Consulta de balance consolidado: El sistema debe permitir al administrador consultar la situación financiera consolidada de todos los perfiles familiares o, si lo desea, visualizar la información individual de un miembro específico mediante un filtro de selección. Adicionalmente el sistema debe permitir al administrador ajustar el rango de fechas o seleccionar una fecha específica para analizar los ingresos y egresos correspondientes.</p>	Alta
RF-03	Gestión de conceptos	<p>Creación de conceptos: El sistema debe permitir al usuario definir el nombre, tipo y periodo del concepto a través de un formulario. Adicionalmente, se incluye el proceso de guardado, que valida los datos ingresados y almacena el nuevo concepto en la base de datos con estado "Habilitado".</p> <p>Edición de conceptos: El sistema debe permitir al usuario modificar la información de un concepto existente, incluyendo su nombre, tipo o periodo, a través de una interfaz de edición. El sistema debe permitir al usuario visualizar los datos actuales del concepto antes de realizar cambios.</p> <p>Deshabilitación de conceptos: El sistema debe permitir al usuario deshabilitar un concepto existente para que no esté disponible al momento de registrar nuevas transacciones, pero manteniendo su historial en el sistema. El sistema debe permitir al usuario confirmar la acción antes de aplicar el cambio de estado.</p>	Alta

Tabla 4.3 Requisitos Funcionales del Sistema

Código	Nombre	Descripción	Prioridad
RF-04	Gestión de perfil	El sistema debe permitir al usuario modificar la información de su perfil de acceso, incluyendo su nombre de visualización y contraseña, mediante una interfaz dedicada a la configuración. El sistema debe permitir al usuario ingresar un nuevo nombre o cambiar su contraseña, validando la contraseña actual y verificando que la nueva cumpla los requisitos de seguridad establecidos.	Alta
RF-05	Ingreso al aplicativo	<p>Registro de cuenta: El sistema debe permitir al usuario crear una cuenta personal mediante un formulario que solicite sus credenciales de acceso y los datos del primer perfil familiar. El sistema debe permitir al usuario registrar un nombre de usuario único y una contraseña para la cuenta, además de definir un perfil inicial con rol administrador. Adicionalmente, se incluye el proceso de validación y guardado, que verifica la unicidad del nombre de usuario y almacena los datos del nuevo usuario y perfil en la base de datos.</p> <p>Inicio de sesión: El sistema debe permitir al usuario acceder al aplicativo ingresando su nombre de usuario y contraseña registrados. El sistema debe permitir la validación de las credenciales y, tras una autenticación exitosa, redirigir al usuario a la pantalla de selección de perfil.</p> <p>Selección de perfil: El sistema debe permitir al usuario, con una sesión ya iniciada, seleccionar un perfil familiar específico para ingresar al entorno principal del sistema. El sistema debe permitir al usuario ingresar la contraseña correspondiente al perfil seleccionado para validar su acceso.</p>	Alta

Tabla 4.4 Requisitos Funcionales del Sistema

Código	Nombre	Descripción	Prioridad
RF-06	Gestión de perfiles familiares	Gestión de perfiles familiares: El sistema debe permitir al usuario con privilegios de administrador crear, editar y eliminar los perfiles de los miembros de un grupo familiar. El sistema debe permitir al administrador crear nuevos perfiles asignando un nombre de usuario único y una contraseña válida; editar los perfiles existentes para otorgar o revocar privilegios de administrador; y eliminar perfiles familiares, asegurando que las transacciones y datos históricos de los miembros eliminados se conserven en el sistema para referencia futura.	Alta
RF-07	Salida del aplicativo	Salida del aplicativo: El sistema debe permitir al usuario finalizar de forma segura su sesión activa. Debe ofrecer la opción de cerrar únicamente la sesión del perfil familiar, redirigiendo a la pantalla de selección de perfiles, o cerrar toda la cuenta, redirigiendo a la pantalla de inicio de sesión. En ambos casos, el sistema debe garantizar la invalidación correcta de la sesión correspondiente para mantener la seguridad y privacidad de los datos del usuario.	Alta

3.2.2 Requisitos No Funcionales

Las siguientes tablas muestran los requisitos no funcionales del sistema FamCash.

Tabla 5.1 Requisitos No Funcionales del Sistema

Código	Nombre	Descripción	Prioridad
RNF-01-Seguridad	Seguridad	El sistema debe implementar medidas de seguridad que garanticen la confidencialidad e integridad de la información en procesos como registro, inicio de sesión, gestión de contraseñas, manejo de saldo y administración de usuarios. Debe cumplir con lo siguiente: - Control de acceso a través de roles de usuario: El sistema debe emplear un esquema de roles (usuario administrador/usuario común) que limite el acceso a funcionalidades o información según el tipo de usuario. - Políticas de contraseñas: Exigir la contraseña actual para realizar el cambio a una nueva.	Alta

Tabla 5.2 Requisitos No Funcionales del Sistema

Código	Nombre	Descripción	Prioridad
RNF-02-Usabilidad	Usabilidad	<p>El sistema debe brindar una experiencia intuitiva, coherente y agradable para los usuarios cumpliendo con:</p> <p>Diseño consistente: Homogeneidad en pantallas (botones, menús, tipografía, colores).</p> <p>Organización clara: Contenidos estructurados (imagen, video, sonido).</p> <p>Facilidad de aprendizaje: Acciones básicas comprensibles y realizables en menos de 2 minutos (registro, inicio, búsqueda, compra, descarga).</p> <p>Manejo de errores: Mensajes claros que expliquen la causa de errores o excepciones.</p>	Alta
RNF-03-Portabilidad	Portabilidad	<p>El sistema debe funcionar correctamente en diversos entornos para garantizar que los usuarios puedan utilizar la aplicación web independientemente del navegador o sistema operativo que usen.</p> <ul style="list-style-type: none"> - Soporte mínimo: El sitio web debe ser funcional en versiones actuales de navegadores modernos (Google Chrome, Mozilla Firefox, Microsoft Edge, Opera y Safari). - Sistemas operativos compatibles: El sitio web debe ser accesible desde Windows, MacOS y Linux a través de navegadores compatibles. 	Alta
RNF-04-Mantenibilidad	Mantenibilidad	<p>El sistema debe estar diseñado de forma que facilite su mantenimiento, permitiendo que las actualizaciones o correcciones se completen en un máximo de 3 días. Debe cumplir con lo siguiente:</p> <ul style="list-style-type: none"> - Comentarios: Mantener comentarios en el código que describen la funcionalidad de cada sección. - Control de versiones: Usar un sistema de control de versiones (git) que permita registrar los cambios y revertir modificaciones en caso de errores. 	Alta

Tabla 5.3 Requisitos No Funcionales del Sistema

Código	Nombre	Descripción	Prioridad
RNF-05-Escalabilidad	Escalabilidad	<p>El sistema debe ser capaz de adaptarse al crecimiento en cuanto a la cantidad de usuarios concurrentes, el volumen de contenidos y el número de categorías, manteniendo tiempos de respuesta eficientes y estables.</p> <p>- Rango estimado: El sistema debe ser capaz de manejar un incremento de 5 a 20 usuarios simultáneos, de 10 a 50 contenidos y de 10 a 100 categorías sin disminuir su rendimiento de manera significativa.</p>	Alta

3.3 Modelo de Requisitos

Representación estructurada de las novedades del usuario.

3.3.1 Modelo de Comportamiento

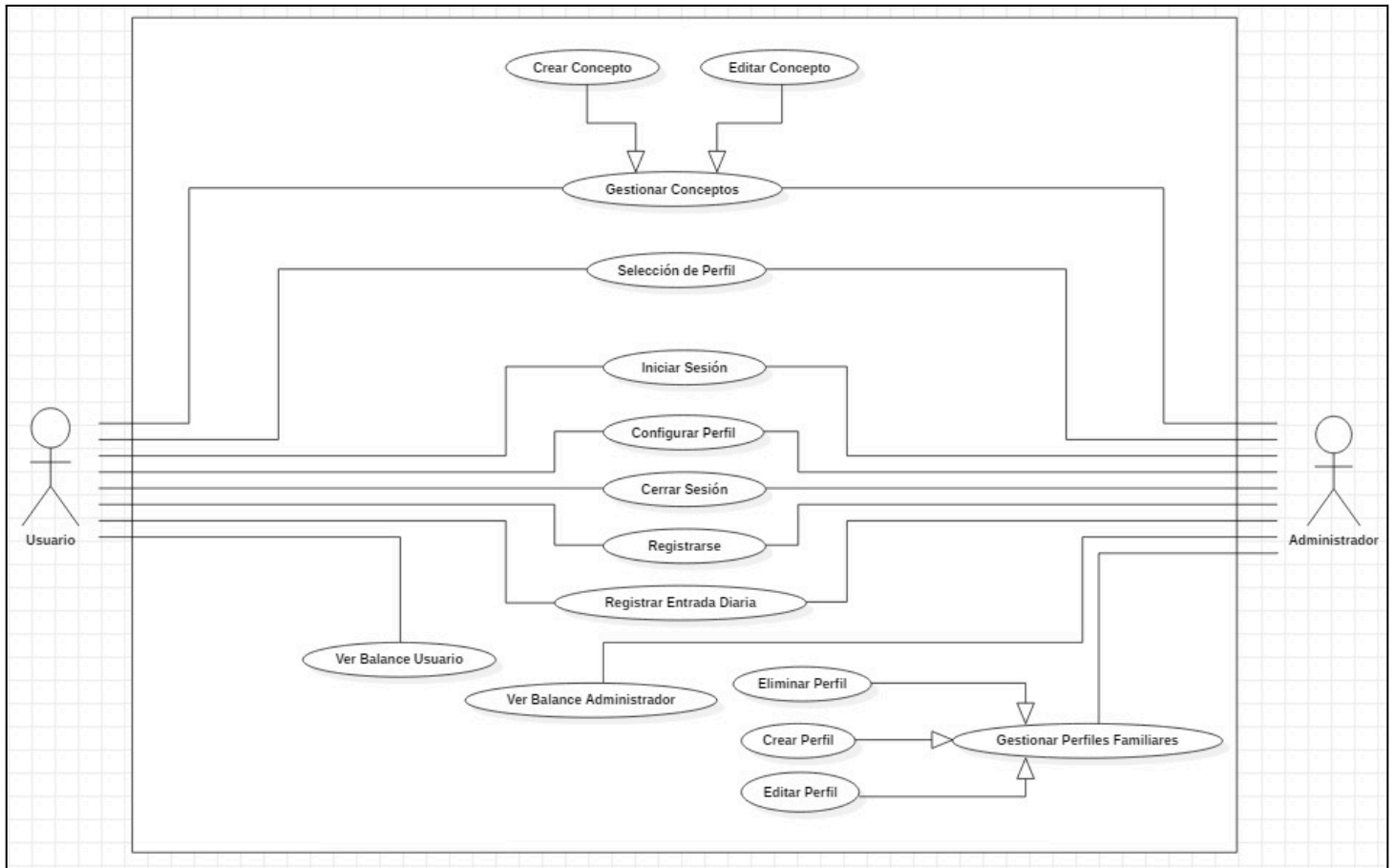
El modelo de comportamiento se representa mediante un modelo de casos de uso y sus especificaciones. Cada caso de uso cuenta con sus especificaciones, donde se explica principalmente el flujo del caso.

3.3.1.a Diagrama de Casos de Uso y Especificación

La Figura 2 muestra el diagrama de casos de uso del sistema FamCash. Este diagrama fue elaborado utilizando la herramienta StarUML.

Se detallan las especificaciones de cada caso de uso del diagrama de casos de uso de la Figura 2. Cada caso de uso tiene su especificación, estos se encuentran en la herramienta al cargar el diagrama de casos de uso.

Figura 2 Diagrama de casos de uso



Link de descarga: <https://github.com/Sawed304/FamCash>

Pasos de ejecución:

- Ingresar al repositorio en Github usando el link proporcionado y descargar el archivo "Diagramas.mdj".
- Abrir el archivo descargado en la herramienta StarUML, para ello si no está instalada, descargar la versión 7.0.
- Selecciona en la pestaña: File ⇒ Open... y escoja el archivo descargado "Diagramas.mdj".
- Dirigirse a "Model Explorer" en la parte derecha y seleccionar "Casos de Uso".
- Para visualizar el diccionario del diagrama de casos de uso, después de seleccionar "Casos de Uso", apretar cada caso de uso individualmente y ver en el área de "Editors" el recuadro de "Documentation".

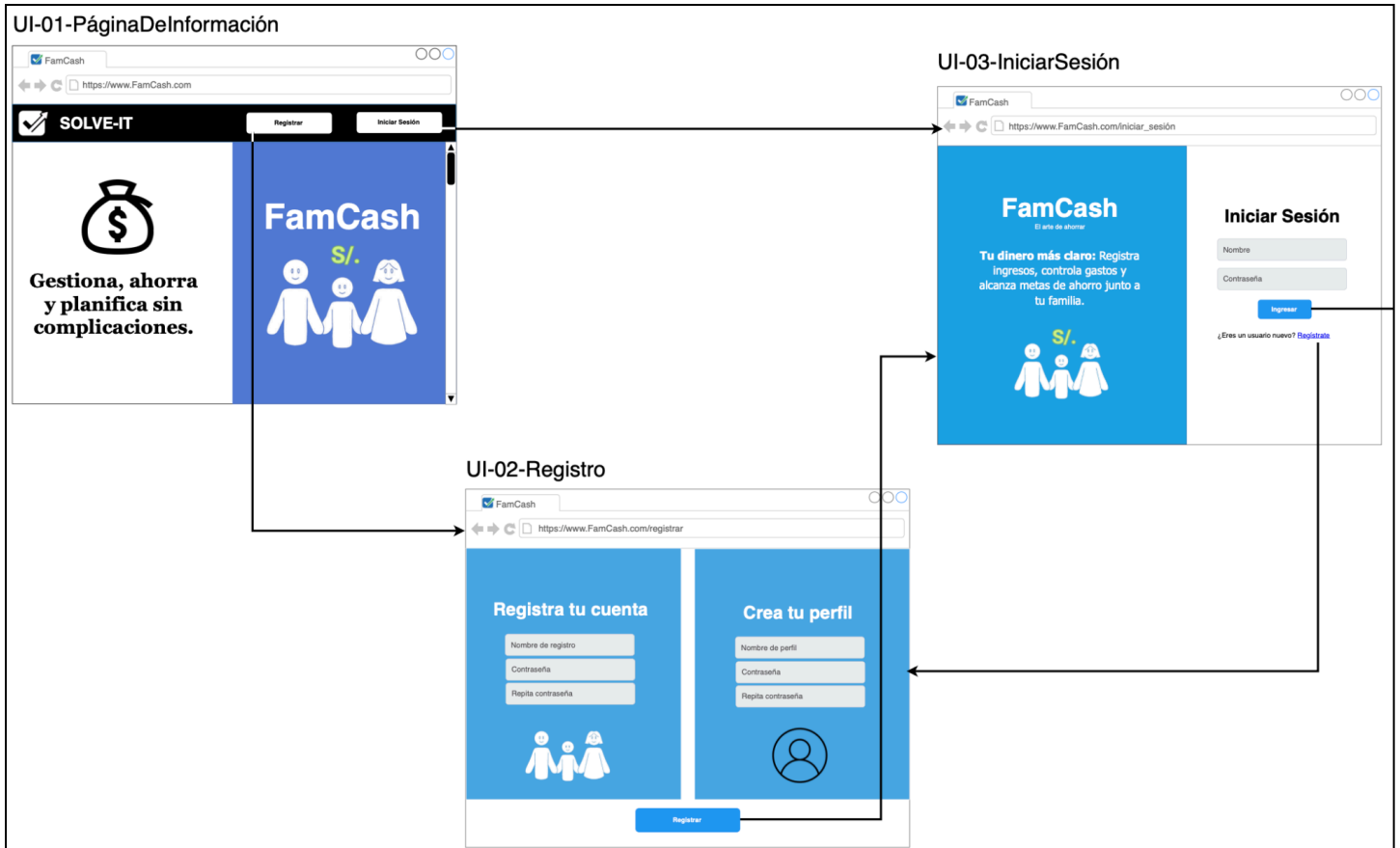
3.3.2 Modelo de Presentación

El modelo de presentación es un diagrama de mapeo, donde se visualizan las interfaces y el mapeo de navegabilidad del sistema FamCash. La interfaz del sistema fue diseñada utilizando la herramienta draw.io.

3.3.2.a Diagrama de Navegabilidad

La Figura 3 muestra los mockups relacionados a los casos de uso CU-001 Registrarse y CU-002 Iniciar Sesión respectivamente.

Figura 3 Mockups caso de uso CU-001 y CU-002



Link de acceso mockup y diccionario: <https://github.com/Sawed304/FamCash>

Pasos de ejecución:

- Ingresar al repositorio en Github usando el link proporcionado y descargar el archivo "mockup.drawio.xml".
- Abrir el archivo descargado en la herramienta draw.io, para ello ingresar a este link <https://app.diagrams.net/>
- Selecciona en la pestaña: Archivo ⇒ Abrir desde ⇒ Dispositivo y escoja el archivo descargado "mockup.drawio.xml".
- Para el diccionario, abrir el link proporcionado y descargar el archivo "Diagrama de Navegabilidad.pdf", abrir el documento y navegar en él usando el índice.

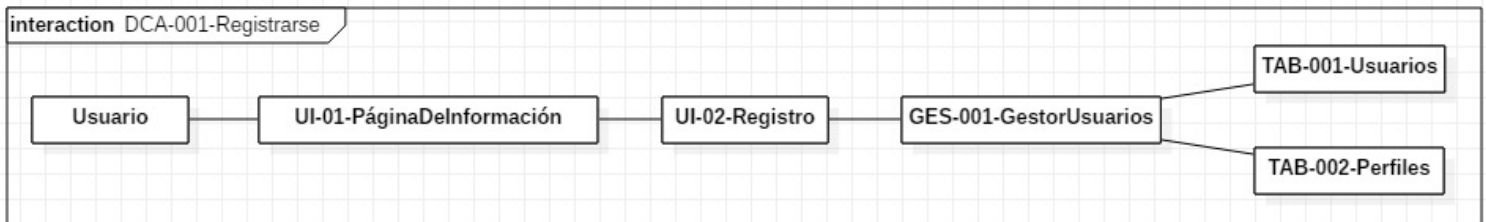
3.3.3 Modelo de Información

El modelo de información ayuda a entender e implementar el comportamiento del sistema FamCash. Los diagramas fueron elaborados utilizando la herramienta StarUML.

3.3.3.a Diagrama de Clases de Análisis

La Figura 4 muestra el diagrama de comunicación DCA-001 correspondiente al caso de uso CU-001 Registrarse.

Figura 4 Diagrama de comunicación DCA-001



Link de descarga: <https://github.com/Sawed304/FamCash>

Pasos de ejecución:

- Ingresar al repositorio en Github usando el link proporcionado y descargar el archivo "Diagramas.mdj".
- Abrir el archivo descargado en la herramienta StarUML, para ello, si esta no está instalada, descargar la versión 7.0.
- Selecciona en la pestaña: File ⇒ Open... y escoja el archivo descargado "Diagramas.mdj".
- Dirigirse a "Model Explorer" en la parte derecha y seleccionar "Diagrama de Clases de Análisis" que es una carpeta que contiene todos los diagramas.
- Para el diccionario, abrir el link proporcionado y descargar el archivo "Diccionario de Clases de Análisis.pdf", abrir el documento y navegar en él usando el índice.

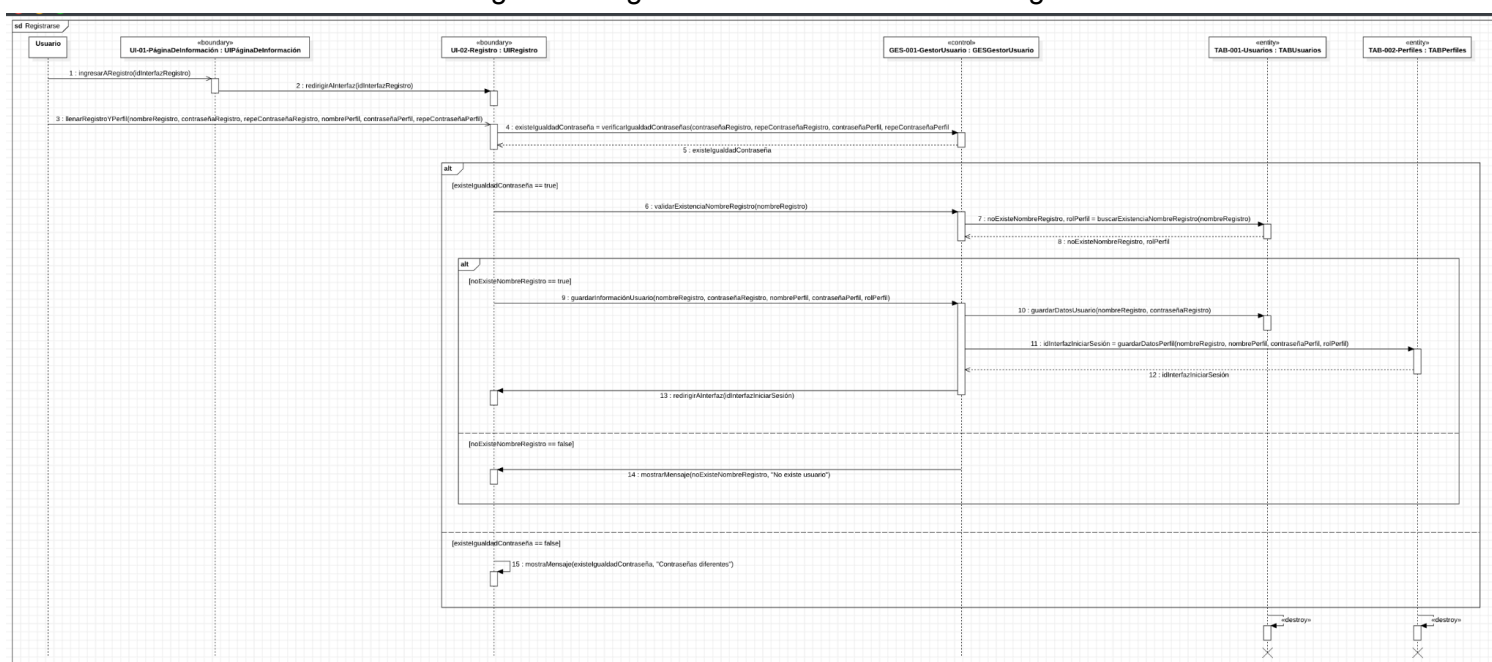
4. Diseño

En esta sección se detalla la fase de diseño de FamCash, que abarca los diagramas de secuencia y los diagramas de clases de análisis. En conjunto, ambos diagramas ofrecen una visión integral que facilita la posterior implementación del sistema.

4.1 Diagramas de Secuencia

En FamCash, los diagramas de secuencia sirven para representar cómo interactúan los usuarios y los distintos módulos del sistema a lo largo de un proceso. Estos diagramas muestran el orden de los mensajes entre actores y objetos, lo que permite comprender la dinámica de cada caso de uso y asegurar que el flujo de información dentro del sistema sea claro y consistente. Los diagramas fueron elaborados utilizando la herramienta StarUML.

Figura 5 Diagrama de secuencia DS-01-Registrarse



Link de descarga: <https://github.com/Sawed304/FamCash>

Pasos de ejecución:

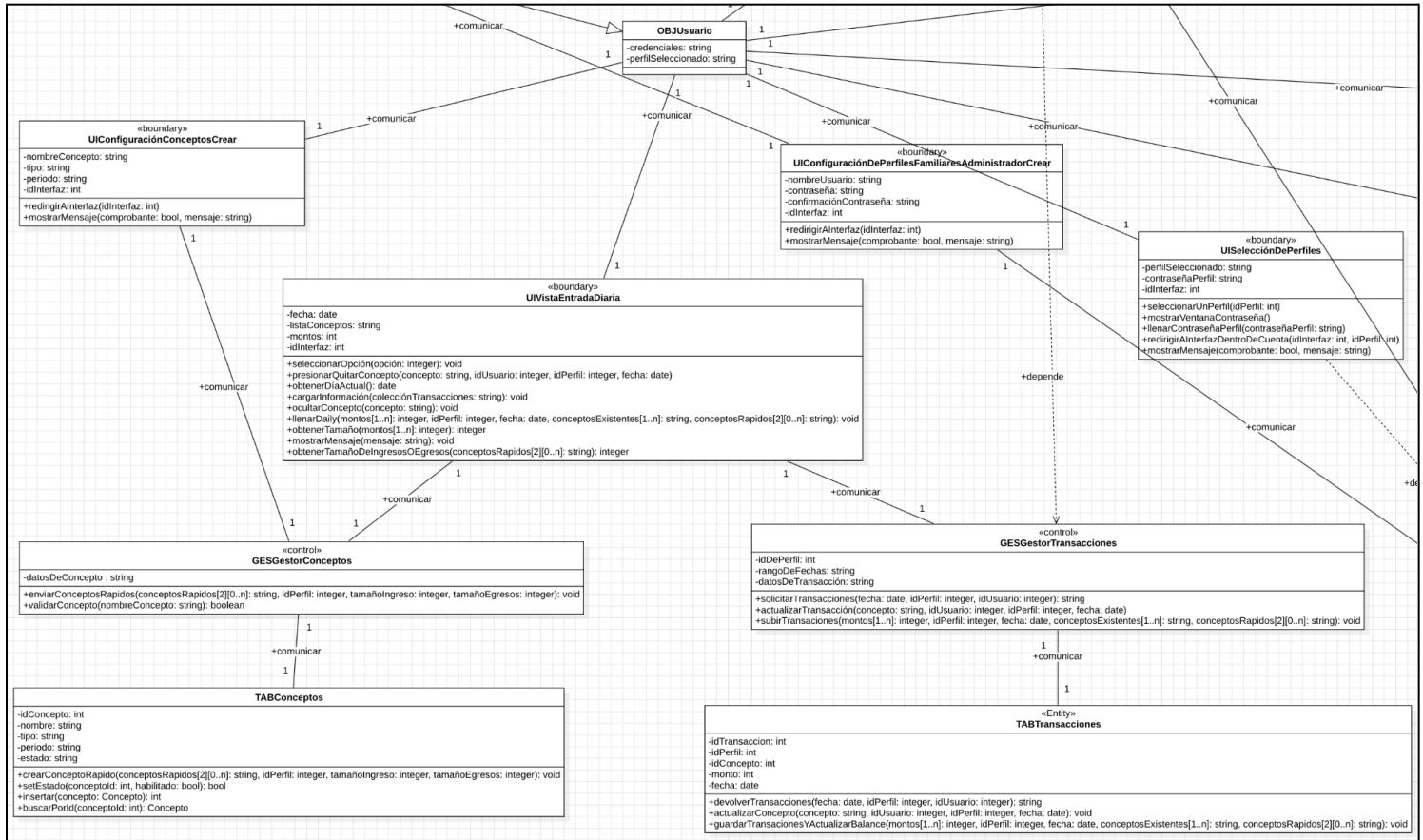
- Ingresar al repositorio en Github usando el link proporcionado y descargar el archivo "Diagramas.mdj".
- Abrir el archivo descargado en la herramienta StarUML, para ello, si esta no está instalada, descargar la versión 7.0.
- Selecciona en la pestaña: File ⇒ Open... y escoja el archivo descargado "Diagramas.mdj".
- Dirigirse a "Model Explorer" en la parte derecha y seleccionar "Diagrama de Secuencia" que es una carpeta que contiene todos los diagramas.

4.2 Diagrama de Clases de Diseño

El diagrama de clases de diseño se abstrae de las relaciones entre los objetos detallados en los diagramas de secuencia y de los mensajes (invocaciones a objetos) con los que se comunicaban.

En la Figura 6 mostramos la sección de Registrar Entrada Diaria.

Figura 6 Diagrama de clases DC-01-DiagramaDeClasesFamCash



Link de descarga: <https://github.com/Sawed304/FamCash>

Pasos de ejecución:

- Ingresar al repositorio en Github usando el link proporcionado y descargar el archivo "Diagramas.mdj".
- Abrir el archivo descargado en la herramienta StarUML, para ello, si esta no está instalada, descargar la versión 7.0.
- Selecciona en la pestaña: File ⇒ Open... y escoja el archivo descargado "Diagramas.mdj".
- Dirigirse a "Model Explorer" en la parte derecha y seleccionar "Diagrama de Clases de Diseño".
- Para el diccionario, abrir el link proporcionado y descargar el archivo "Diccionario de Clases de Diseño.pdf", abrir el documento y navegar en él usando el índice.

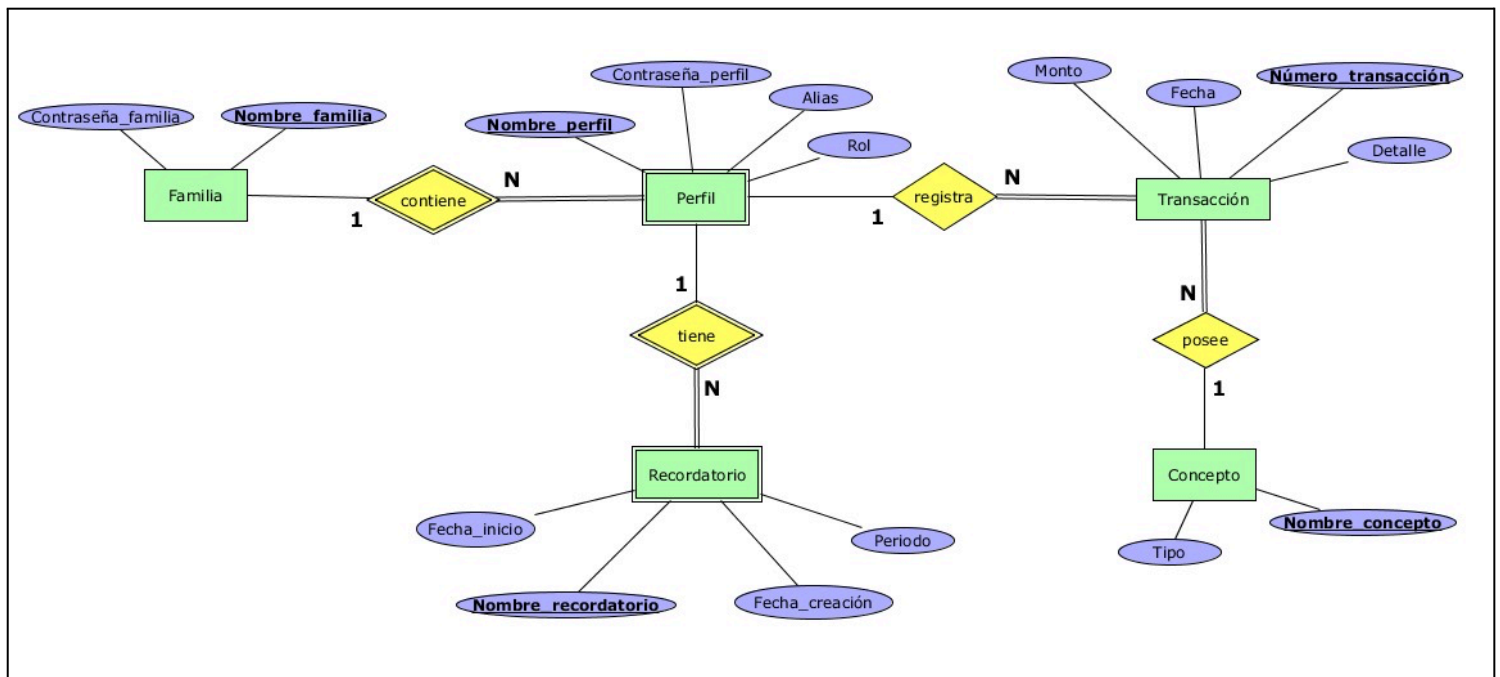
4.3 Modelo Entidad-Relación

El modelo entidad-relación describe la estructura lógica del sistema FamCash, identificando las principales entidades, sus atributos y las relaciones que existen entre ellas.

4.3.1 Diagrama del Modelo Entidad-Relación

La Figura 7 presenta el diagrama del modelo entidad-relación del sistema FamCash. Este modelo sirve como base para el posterior diseño de la base de datos relacional. Este modelo fue elaborado utilizando la herramienta TerraER.

Figura 7 Diagrama del modelo entidad-relación (MER-001)



Link de descarga: <https://github.com/Sawed304/FamCash>

Pasos de ejecución:

- Ingresar al repositorio en Github usando el link proporcionado y descargar el archivo "MER-001.xml".
- Abrir el archivo descargado en la herramienta TerraER, para ello si no está instalada, descargar la versión 3.14.
- Selecciona en la pestaña: File \Rightarrow open y escoja el archivo descargado "MER-001.xml".
- Para el diccionario, abrir el link proporcionado y descargar el archivo "Diccionario del Modelo Entidad-Relación.pdf", abrir el documento y navegar en él usando el índice.

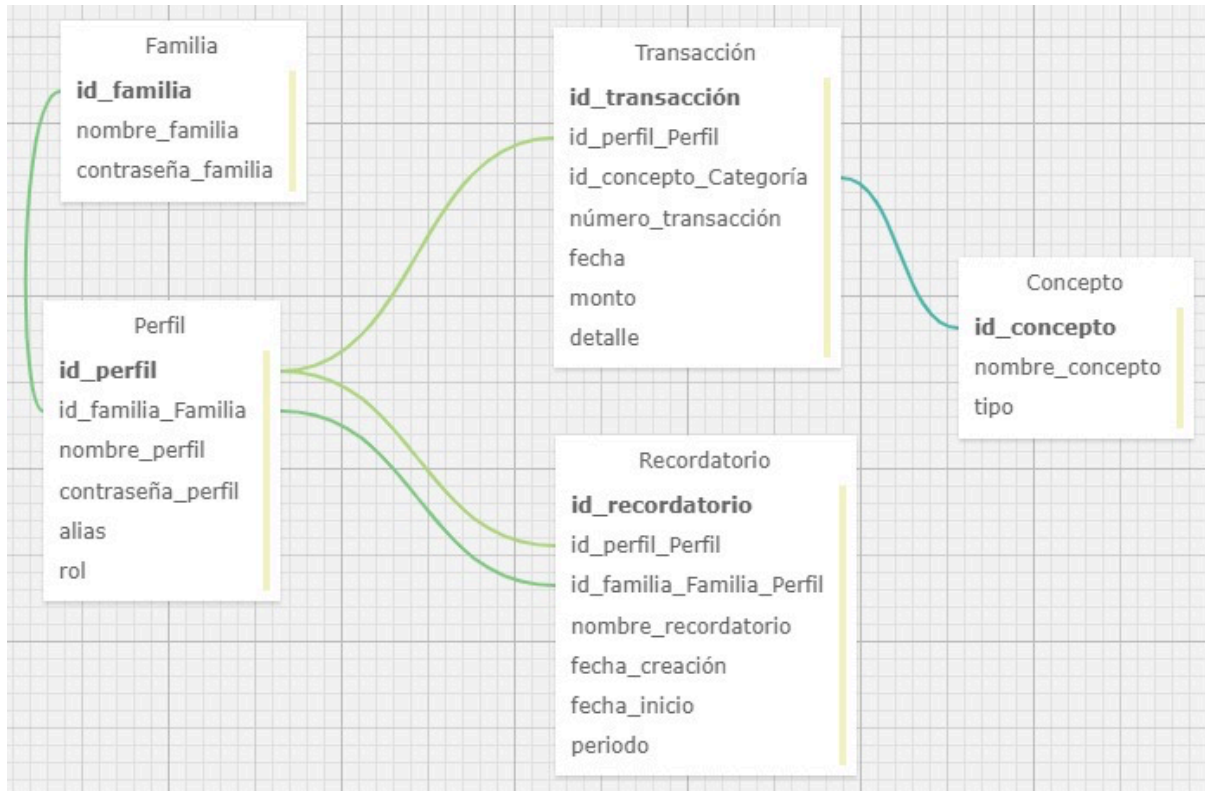
4.4 Modelo Relacional

El modelo relacional representa la estructura lógica de la base de datos del sistema FamCash. Define las entidades, atributos y relaciones que permiten almacenar y gestionar la información del sistema de manera estructurada.

4.4.1 Diagrama del Modelo Relacional

La Figura 8 muestra el diagrama del modelo relacional del sistema FamCash. Este diagrama fue elaborado utilizando la herramienta SQL Designer.

Figura 8 Diagrama del modelo relacional (MR-001)



Link de descarga: <https://github.com/Sawed304/FamCash>

Pasos de ejecución:

- Ingresar al repositorio en Github usando el link proporcionado y descargar el archivo "MR-001.txt".
- Abrir el archivo descargado (block de notas).
- Abrir la herramienta SQL Designer, para ello ingresar a este link <https://sql.toad.cz/>
- Selecciona en la pestaña: Save / Load ⇒ Copiar el texto del archivo "MR-001.txt" ⇒ Pegar el texto en la parte Input / Output ⇒ Presionar LOAD XML de la parte Client ⇒ Presionar ok.
- Para el diccionario, abrir el link proporcionado y descargar el archivo "Diccionario del Modelo Relacional.pdf", abrir el documento y navegar en él usando el índice.

4.5 Plan de Pruebas

El plan de pruebas del sistema FamCash establece el alcance de las pruebas, tipos de pruebas a ejecutar, los diseños de casos de prueba, los criterios de aceptación, los responsables, el cronograma asociado y los recursos involucrados en el proceso de validación y verificación del sistema.

Link de descarga: <https://github.com/Sawed304/FamCash>

Pasos de ejecución:

- Ingresar al repositorio en Github usando el link proporcionado y descargar el archivo "Plan de Pruebas.pdf".
- Abrir el documento y navegar en él usando el índice.

5. Construcción

En esta sección se presenta la fase de construcción de FamCash, donde se implementa el sistema según los modelos de diseño. Incluye el desarrollo del código, la integración de módulos y la verificación de su correcto funcionamiento.

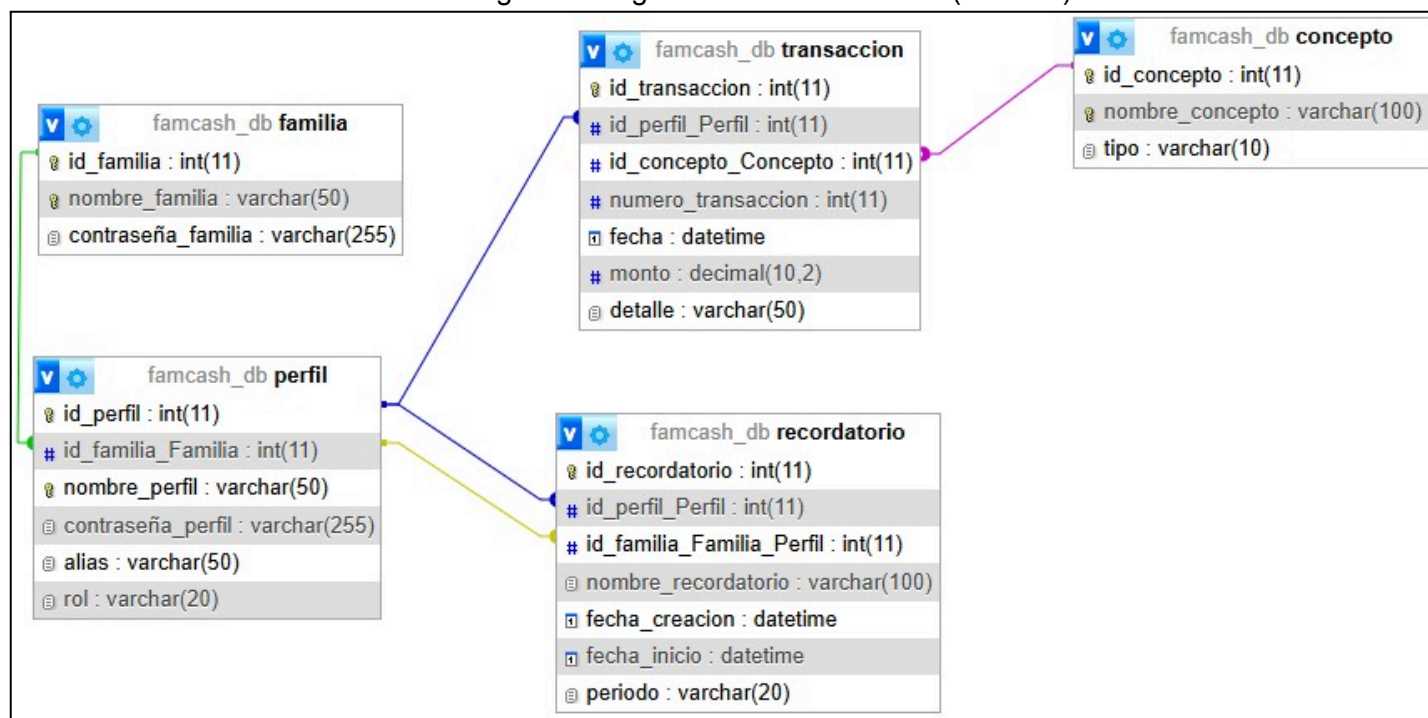
5.1 Modelo Físico

El modelo físico describe la implementación concreta de la base de datos del sistema FamCash en el sistema gestor de base de datos elegido (phpMyAdmin con MySQL).

5.1.1 Diagrama del Modelo Físico

La Figura 9 presenta el diagrama del modelo físico de la base de datos de FamCash, incluyendo los tipos de datos definidos para su implementación en MySQL.

Figura 9 Diagrama del modelo físico (MF-001)



Link de descarga: <https://github.com/Sawed304/FamCash>

Pasos de ejecución:

- Ingresar al repositorio en Github usando el link proporcionado y descargar el archivo "MF-001.sql".
- Abrir el archivo descargado en la herramienta phpMyAdmin mediante XAMPP.
- Crear una base de datos con el mismo nombre que el archivo descargado.
- Selecciona en la pestaña: Importar ⇒ Elegir archivo ⇒ Escoger el archivo descargado "MF-001.sql" ⇒ Presionar la BD ⇒ Ir a "Más" ⇒ Diseñador.
- Para el diccionario, abrir el link proporcionado y descargar el archivo "Diccionario del Modelo Físico.pdf", abrir el documento y navegar en él usando el índice.

5.1.2 Proyección de Crecimiento

A continuación se realizará una proyección de crecimiento de la base de datos de FamCash con el objetivo de planificar el aumento del almacenamiento para mantener la operatividad del sistema.

Supuestos iniciales

- **Familias (Cuentas) registradas:** 50 (lanzamiento inicial).
- **Usuarios (Perfiles) promedio por familia:** 3 perfiles (ej. "Papá", "Mamá", "Hijo").
- **Total de Usuarios (Perfiles) iniciales:** 150.
- **Actividad estimada por mes:**
 - ❖ Se asume que cada perfil (Usuario) registra un promedio de 2 transacciones diarias (ingresos o egresos), como se ve en la "Entrada diaria".
 - ❖ **Transacciones mensuales:** 150 perfiles x 2 trans./día x 30 días = 9,000 transacciones/mes.
- **Tamaño medio de un registro de transacción:** Estimado en 1 KB. Esta es una estimación generosa que incluye la fila en la tabla Transacciones, las filas asociadas en tablas de unión (para Conjunto_conceptos), índices de la base de datos y sobrecarga de gestión.
- **Crecimiento estimado:** 15% mensual (compuesto por nuevas familias, nuevos perfiles y el incremento de registros de actividad).
- **Volumen de almacenamiento inicial:** SSD de 20 GB.

La Tabla 6 estima el espacio ocupado en las condiciones iniciales (mes 0).

Tabla 6 Estimación del espacio ocupado

Tabla / datos	Espacio
Metadatos (Todas las Tablas)	1 MB aproximado
Datos de Actividad (9,000 trans. x 1 KB)	9 MB
TOTAL BD (Mes 0)	10 MB (0.01 GB)

Sean $S_0 = 0.01$ GB el tamaño inicial y $S(n) = S_0 \times (1.15)^n$ el tamaño tras n meses.

La tabla 7 muestra lo que es la proyección de ocupación del volumen de 20 GB con un crecimiento del 15 % mensual.

Tabla 7 Proyección de ocupación del volumen de 20 GB con un crecimiento del 15 % mensual

Mes n	S(n) (GB)	% de 20 GB
0	0.01	0.05 %
12	$0.01 \times 1.15^n = 0.05$	0.3 %
24	0.29	1.5 %
36	1.53	7.7 %
48	8.19	41.0 %
53	14.30	71.5 %
54	16.44	82.2 %
55	18.91	94.6 %
56	21.75	108.8 %

Con la información de la tabla 7 se pueden llegar a las siguientes conclusiones:

- Con un disco de 20 GB, el umbral del 100 % se alcanzaría alrededor del mes 56 (cuatro años y ocho meses) suponiendo un crecimiento sostenido del 15 % mensual.
- Para conservar un margen operativo del 20 % (llegar al 80 % de capacidad), la ampliación del almacenamiento debería planificarse hacia el mes 54 (cuatro años y medio).

5.1.3 Copias de Seguridad

Para realizar las copias de seguridad de la base de datos del sistema FamCash se realizarán las siguientes acciones:

- Copia completa diaria (madrugada 03:00) usando `pg_dumpall --format=custom`. Dada la naturaleza transaccional del sistema, se requiere una copia diaria para minimizar la pérdida de datos (RPO de 24 horas).
- Sincronización en la nube una vez por día, inmediatamente tras la copia completa, con `rclone` a un bucket S3.
- Retención
 - Últimas 7 copias completas locales: 7 días.
 - Copias en nube: 3 meses.

La tabla 8 resume el calendario de los backups a realizarse.

Tabla 8 Calendario resumido de backups

Componente	Tipo	Frecuencia	Herramienta
Base de datos	Completa	Diaria (mad.)	pg_dumpall -Fc
Off-site	Sincronización	Diaria (mad.)	rclone a S3

5.1.4 Implementación del Modelo Físico

A continuación, se presenta la implementación del modelo físico (MF-001) de la base de datos del sistema FamCash.

Link de descarga: <https://github.com/Sawed304/FamCash>

Pasos de ejecución:

- Ingresar al repositorio en Github usando el link proporcionado e ingresar a la carpeta “Aplicativo”.
- Descargar el archivo “famcash_db” de la carpeta.
- Para cargar la base de datos, debemos tener instalado “XAMPP” en nuestra computadora.
- Al abrirlo, dar click en “Start” a los apartados de “Apache” y “MySQL”. Luego presionar “Admin” de la parte de MySQL y se nos abrirá una pestaña de Chrome de “phpMyAdmin”.
- Presionar “Nueva” situada en el lateral izquierdo, en el recuadro “Nombre de la base de datos” escribir “famcash_db” y darle en “crear”.
- Luego hacemos click en la base de datos creada y en la parte superior presionamos “Importar”, finalmente cargamos el archivo descargado “famcash_db.sql” y le damos a importar.

5.2 Producto

Esta sección reúne la evidencia tangible del avance del proyecto. A continuación se proporciona el enlace al repositorio oficial donde reside el código fuente y la base de datos.

Link de descarga: <https://github.com/Sawed304/FamCash>

Pasos de ejecución código fuente:

- Ingresar al repositorio en Github usando el link proporcionado e ingresar a la carpeta “Aplicativo”.
- Descargar el archivo “FAMCASH.zip” y descomprimirlo.
- Para el correcto funcionamiento del código fuente, nos dirigimos al disco donde se encuentra instalado “XAMPP” (generalmente el disco C).
- Ubicamos la carpeta “xampp” e ingresamos, dentro ubicamos la carpeta “htdocs” y ahí pegamos la carpeta descomprimida.
- Para abrir el código, en su entorno de programación de preferencia (recomendable usar Visual Studio Code) abrir la carpeta descomprimida “FAMCASH” que se encuentra en la carpeta htdocs.

6. Referencias

Object Management Group. Unified Modeling Language (UML).
<https://www.omg.org/spec/UML/>

MKLab Co., Ltd. StarUML – A sophisticated software modeler for agile and concise modeling. <https://staruml.io/>