

EASYFINANCE

EF

Autor: Héctor Santana Camacho

Fecha: 21/12/2025

Versión 1.0



Diagrama Entidad-Relación

EasyFinance

DAW. Desarrollo de Aplicaciones Web
PWR. Proyecto de Desarrollo de Aplicaciones Web

Nombre del fichero:	DAW_PRW_EF_UT01.3. Diagrama_ER.pdf
Fecha de esta versión:	18/01/2026



Historial de revisiones

Fecha	Descripción	Autor
18/01/2026	Creación del documento	Héctor Santana Camacho

ÍNDICE

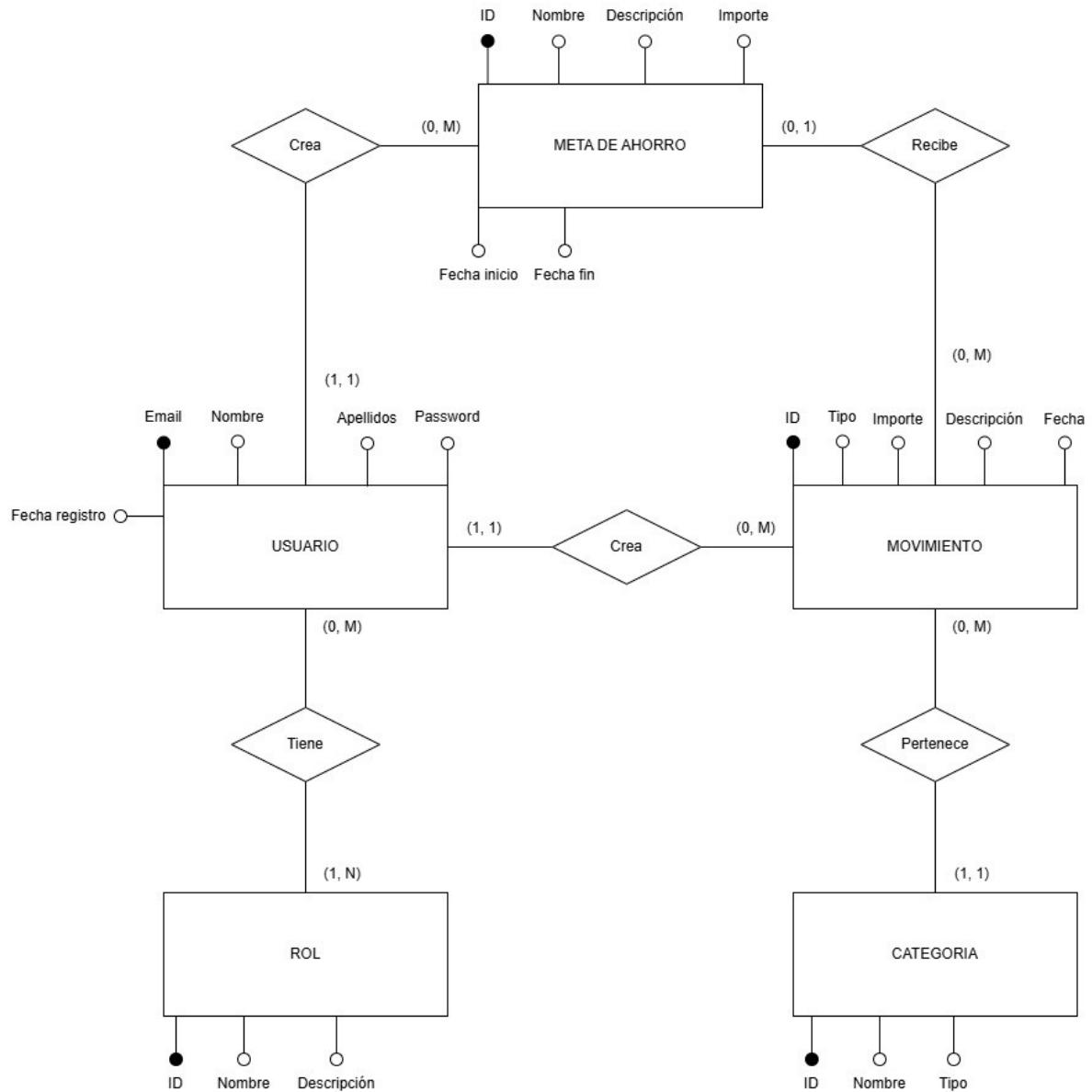
1 Introducción.....	4
2 Diagrama Entidad-Relación.....	4
2.1 Descripción de Entidades y Atributos.....	5
2.1.1 USUARIO.....	5
2.1.2 MOVIMIENTO.....	6
2.1.3 CATEGORIA.....	6
2.1.4 META DE AHORRO.....	7
2.1.5 ROL.....	7
3 Justificación de Decisiones de Diseño.....	7
3.1 Elección de claves primarias.....	8
3.2 Cardinalidades y restricciones de integridad.....	8
3.3 Normalización y evitación de redundancia.....	8
4 Conclusión.....	8

1 Introducción

El presente documento describe el modelo de datos que sustenta la aplicación, representado mediante un Diagrama Entidad-Relación (E/R) que define las entidades principales del sistema, sus atributos clave y las relaciones que garantizan la integridad referencial y funcional de la base de datos.

2 Diagrama Entidad-Relación

A continuación se presenta el diagrama E/R completo del sistema EasyFinance, elaborado siguiendo la notación de Chen, que representa las entidades mediante rectángulos, las relaciones mediante rombos, y los atributos mediante óvalos conectados a sus respectivas entidades:



Notación utilizada: Modelo de Chen (Peter Chen, 1976)

Rectángulo: representa una entidad del sistema

Rombo: representa una relación entre entidades

Óvalo: representa un atributo de una entidad

Óvalo relleno (●): indica atributo clave primaria (identificador único)

Cardinalidades: se expresan junto a las líneas de relación como (mínimo, máximo), donde los valores pueden ser 0, 1, N o M.

2.1 Descripción de Entidades y Atributos

2.1.1 USUARIO

Representa a las personas registradas en el sistema que utilizan la aplicación para gestionar sus finanzas personales.

Atributo	Tipo	Descripción	Restricciones
Email	VARCHAR(100)	Correo electrónico del usuario (clave primaria)	PK, NOT NULL, UNIQUE
Nombre	VARCHAR(50)	Nombre del usuario	NOT NULL
Apellidos	VARCHAR(100)	Apellidos del usuario	NOT NULL
Password	VARCHAR(255)	Contraseña cifrada (hash BCrypt)	NOT NULL
Fecha_registro	DATE	Fecha de alta en el sistema	NOT NULL, DEFAULT CURRENT_DATE

Relaciones:

Un usuario puede crear de 0 a muchos (0, M) movimientos financieros.

Un usuario puede crear de 0 a muchas (0, M) metas de ahorro.

Un usuario tiene de 1 a muchos (1, N) roles.

2.1.2 MOVIMIENTO

Representa las transacciones financieras (ingresos o gastos) registradas por el usuario.

Atributo	Tipo	Descripción	Restricciones
ID	BIGINT	Identificador único del movimiento	PK, AUTO_INCREMENT, NOT NULL
Tipo	ENUM('INGRESO', 'GASTO')	Tipo de movimiento	NOT NULL
Importe	DECIMAL(10,2)	Cantidad monetaria del movimiento	NOT NULL, > 0
Descripción	VARCHAR(255)	Descripción detallada del movimiento	NULL
Fecha	DATE	Fecha en la que se realizó el movimiento	NOT NULL

Relaciones:

Un movimiento pertenece a una categoría (1, 1).

Un movimiento es creado por 1 usuario (1, 1).

Un movimiento puede ser recibido por 0 o 1 meta de ahorro (0, 1).

2.1.3 CATEGORIA

Define las clasificaciones de los movimientos financieros (Salario, Alquiler, Alimentación, Transporte, etc.).

Atributo	Tipo	Descripción	Restricciones
ID	BIGINT	Identificador único de la categoría	PK, AUTO_INCREMENT, NOT NULL
Nombre	VARCHAR(50)	Nombre de la categoría	NOT NULL, UNIQUE
Tipo	ENUM('INGRESO', 'GASTO')	Tipo de categoría	NOT NULL

Relaciones:

Una categoría puede clasificar de 0 a muchos movimientos (0, M).

2.1.4 META DE AHORRO

Representa los objetivos de ahorro personales establecidos por el usuario.

Atributo	Tipo	Descripción	Restricciones
ID	BIGINT	Identificador único de la meta	PK, AUTO_INCREMENT, NOT NULL
Nombre	VARCHAR(100)	Nombre descriptivo de la meta	NOT NULL
Descripción	VARCHAR(255)	Descripción detallada del objetivo	NULL
Importe	DECIMAL(10,2)	Meta de ahorro a alcanzar	NOT NULL, > 0
Fecha_inicio	DATE	Fecha de creación de la meta	NOT NULL
Fecha_fin	DATE	Fecha límite para alcanzar la meta	NULL

Relaciones:

Una meta de ahorro es creada por un usuario (1, 1).

Una meta de ahorro puede recibir 0 o muchos aportes/ingresos registrados (0, M).

2.1.5 ROL

Define los roles de usuario dentro del sistema (Usuario estándar, Administrador) para gestión de permisos.

Atributo	Tipo	Descripción	Restricciones
ID	BIGINT	Identificador único del rol	PK, AUTO_INCREMENT, NOT NULL
Nombre	VARCHAR(50)	Nombre del rol (USER, ADMIN)	NOT NULL, UNIQUE
Descripción	VARCHAR(255)	Descripción de los permisos del rol	NULL

Relaciones:

Un rol puede ser asignado a 0 o muchos usuarios (0, M).

3 Justificación de Decisiones de Diseño

El diseño del modelo de datos se ha estructurado siguiendo los principios de normalización relacional y las buenas prácticas de diseño de bases de datos, garantizando escalabilidad, integridad referencial y alineación con los requisitos funcionales del proyecto EasyFinance.

3.1 Elección de claves primarias

Se ha optado por utilizar el email como clave primaria en la entidad USUARIO, dado que es un atributo único, inmutable y significativo que facilita la autenticación y la gestión de sesiones. Para el resto de entidades (MOVIMIENTO, CATEGORIA, META DE AHORRO, ROL), se emplean claves primarias artificiales autoincrementales (ID), siguiendo el estándar para entidades de negocio, lo que garantiza unicidad, eficiencia en las consultas y facilita las relaciones entre tablas.

3.2 Cardinalidades y restricciones de integridad

Las cardinalidades definidas responden directamente a los requisitos funcionales del sistema. Por ejemplo, la relación (1, 1) a (0, M) entre USUARIO y MOVIMIENTO refleja que todos los movimientos deben pertenecer a un usuario (para garantizar la seguridad y privacidad de los datos financieros), mientras que un usuario puede no tener movimientos registrados inicialmente.

3.3 Normalización y prevención de redundancia

El modelo cumple con la Tercera Forma Normal (3FN), eliminando dependencias transitivas y redundancia de datos. Por ejemplo, los atributos relacionados con categorías no se repiten en la tabla MOVIMIENTO; en su lugar, se establece una clave foránea hacia CATEGORIA, lo que asegura consistencia y facilita actualizaciones centralizadas (si se cambia el nombre de una categoría, el cambio se refleja automáticamente en todos los movimientos asociados).

4 Conclusión

El Diagrama Entidad-Relación presentado, elaborado siguiendo la notación de Chen, constituye una representación completa, coherente y técnicamente sólida del modelo de datos de EasyFinance. El diseño responde de forma precisa a los requisitos funcionales del sistema (gestión de usuarios, movimientos, metas de ahorro, categorías y roles) y garantiza la integridad referencial mediante cardinalidades y restricciones bien definidas.

Este modelo es escalable y flexible, permitiendo futuras ampliaciones del sistema sin necesidad de reestructurar la base de datos. Además, facilita la comprensión técnica del proyecto para su implementación práctica en las siguientes fases de desarrollo.