Database Design Document (DDD)

Version 1.0

Prepared by: Paulos, Rae S.
Marisga, Jersey Mae D.
Anastacio, Carl Zeus n
Pilapil, Jared L.

Revision History

Date	Version	Description	Author

Table of Contents

1 Introduction

The section introduces the Database Design Document (DDD) for the GCash application to its readers.

1.1 Document Objectives

This DDD for GCash application has the following objectives:

- Provide a comprehensive description of tables, fields, and constraints necessary for secure and efficient data management
- Define the structure and relationships of the MySQL database supporting the GCash application
- To serve as the basis for implementing the database.

1.2 Intended Audiences

This DDD is intended for the following audiences:

- Technical reviewers and supervisors who must evaluate the quality of this document.
- Developers including:
  Designers, to confirm that the database supports UI/UX needs
  Programmers, to implement database functions using JavaFX and JDBC
  Tester, to check if the database performs correctly.

1.3 References

This DDD refers to the following references:

- Project Specifications: https://believed-bongo-319.notion.site/CTINFMGL-Project-Specifications-19296450aad180aea9ebf27987415f4c

2 Detailed Database Design

This section describes the actual design of different databases at varying levels of abstraction. A subsection for each of the conceptual, internal, logical, and physical levels.

2.1.1 Data dictionary

2.1.1.1 Data dictionary for Element: ADMIN

Name	**Data Type**	**constraint**	**Description**
admin_ID	int AI	primary key, auto-increment	Unique identifier for the admin.
admin_full_name	varchar (50)	not null	Full name of the admin
admin_email_address	varchar (50)	not null, unique	Email address of the admin.
admin_PIN	char(4)	not null, check (admin_PIN BETWEEN '0000' AND '9999')	Personal Identification Number of the admin.

2.1.1.2 Data dictionary for Element: USER

Name	**Data Type**	**constraint**	**Description**
phone_number (primary key)	varchar (15)	primary key, not null, unique	Phone number and unique identifier for the user.
first_name	varchar (50)	not null	First name of the user.
last_name	varchar (50)	not null	Last name of the user.
email_address	varchar (50)	not null, unique	Email address of the user.
PIN	Integer (4)	not null, check (PIN~ '^[0-9]{4}$'))	Personal Identification Number of the user.
birthdate	Date (mm-dd-yyyy)	not null, check (birthdate <= current_date)	Date of birth of the user.
country	varchar (20)	not null	Country of residence of the user.
address	varchar (50)	not null	Residential address of the user.

2.1.1.3 Data dictionary for Element: send_transactions

Name	Data Type	constraint	Description
send_transaction_ID (primary key)	Integer (Auto increment)	primary key, auto increment	Unique identifier for each transaction.
sender_number	varchar (15)	not null	Phone number of sender.
sender_name	varchar (50)	not null	Name of the sender.
receiver_number	Integer (11)	not null	Phone number of receiver
receiver_name	varchar (50)	not null	Name of the receiver.
amount	float (10, 2)	not null, check (amount > 0)	Amount of money being sent in the transaction.
transaction_date	Date (mm-dd-yyyy)	not null, check (transaction_date <= current_timestamp)	The date when the transaction was made.

2.1.1.4 Data dictionary for Element: deposit_transactions

Name	Data Type	constraint	Description
deposit_transaction_ID (primary key)	Integer (Auto increment)	primary key, auto-increment	Unique identifier for each deposit transaction.
deposit_number	varchar (15)	not null, unique	A unique number assigned to each deposit
depositor_name	varchar (50)	not null	The name of the person making the deposit
amount	float (10, 2)	not null, check (amount > 0)	The amount of money deposited.
transaction_date	Date (mm-dd-yyyy)	not null, check (transaction_date <= current_timestamp)	The date when the deposit transaction was recorded.

2.1.1.5 Data dictionary for Element: withdraw_transactions

Name	**Data Type**	**constraint**	**Description**
withdraw_transaction_ID(primary key)	Integer (Auto increment)	primary key, auto increment	Unique identifier for each withdrawal transaction.
withdrawer_number	Integer (15)	not null	A numerical identifier assigned to the individual making the withdrawal.
withdrawer_name	varchar (50)	not null	Name of the person making the withdrawal.
amount	float (10, 2)	not null, check (amount >0)	The amount of money withdrawn (must be greater than zero).
transaction_date	Date (mm-dd-yyyy)	not null, check (transaction_date <= current timestamp)	The date when the withdrawal transaction was recorded.

2.1.1.6 Data dictionary for Element: WALLET

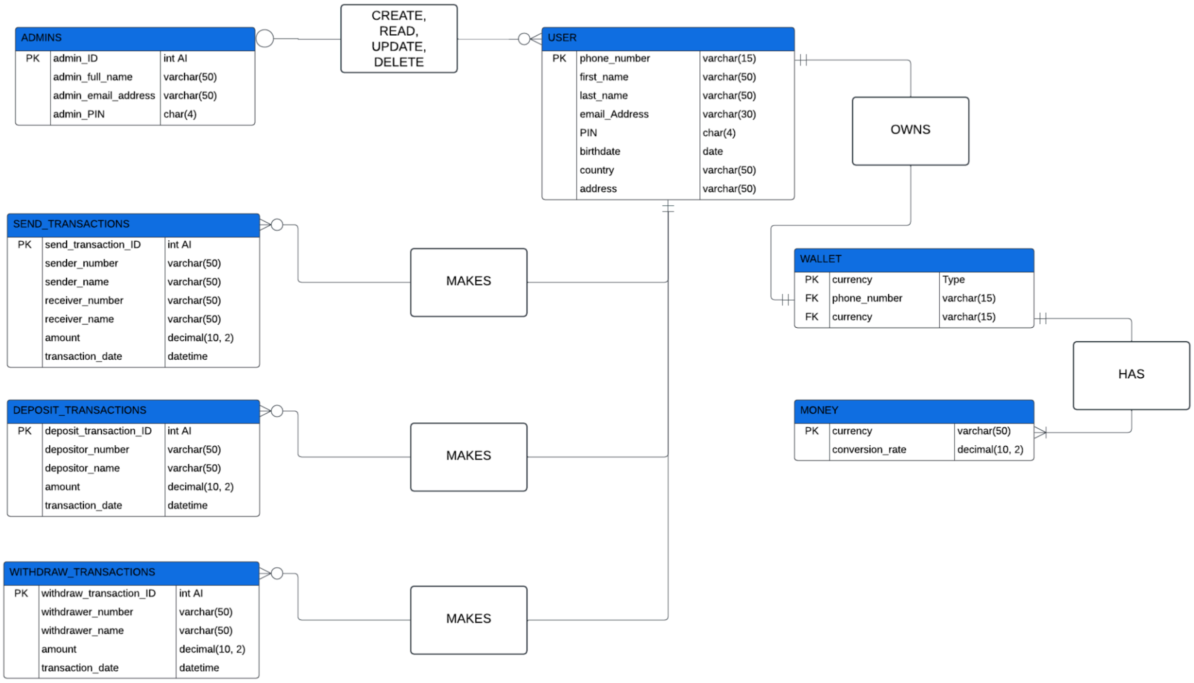
Name	**Data Type**	**constraint**	**Description**
phone_number (primary key)	Integer (15)	primary key, not null	Inique identifier assigned to each wallet user.
first_name	varchar (50)	not null	The first name of the user
last_name	varchar (50)	not null	The last name of the user
balance	float (10, 2)	not null	The current amount of money stored in the wallet.
currency	String	not null	The currency type.

2.1.1.7 Data dictionary for Element: MONEY

Name	**Data Type**	**constraint**	**Description**
currency (primary key)	String	primary key, not null	Represents currency code
conversion_rate	float (10, 2)	not null, check (conversion_rate > 0)	Defines the currency's conversion rate relative to a base currency (e.g., USD).

2.2 MySQL database design (Relational database)

2.2.1 Conceptual diagram


ADMINS
PK admin_ID int AI
admin_full_name varchar(50)
admin_email_address varchar(50)
admin_PIN char(4)
CREATE, READ, UPDATE, DELETE
USER
PK phone_number varchar(15)
first_name varchar(50)
last_name varchar(50)
email_Address varchar(30)
PIN char(4)
birthdate date
country varchar(50)
address varchar(50)
OWNS
SEND_TRANSACTIONS
PK send_transaction_ID int AI
sender_number varchar(50)
sender_name varchar(50)
receiver_number varchar(50)
receiver_name varchar(50)
amount decimal(10, 2)
transaction_date datetime
MAKES
WALLET
PK currency Type
FK phone_number varchar(15)
FK currency varchar(15)
HAS
DEPOSIT_TRANSACTIONS
PK deposit_transaction_ID int AI
depositor_number varchar(50)
depositor_name varchar(50)
amount decimal(10, 2)
transaction_date datetime
MAKES
MONEY
PK currency varchar(50)
conversion_rate decimal(10, 2)
WITHDRAW_TRANSACTIONS
PK withdraw_transaction_ID int AI
withdrawer_number varchar(50)
withdrawer_name varchar(50)
amount decimal(10, 2)
transaction_date datetime
MAKES

2.2.2 Description

TThis conceptual diagram represents the structure of a GCash application. To use the system, users must first sign up by providing their first name, last name, email address, PIN, birthdate, country, and address. Once registered, they can log in using their phone number and PIN as credentials. Each user automatically owns a wallet upon registration, which starts with a default balance of ₱0 and uses PHP as the standard currency. Users can perform essential financial transactions, including sending money to other registered users, depositing funds into their wallets, and withdrawing money when needed. All transactions are recorded and can be viewed in the transaction history, allowing users to track their financial activities efficiently.

2.2.2.1 Purpose of ADMIN Table

This table stores the application's database. The administrator has exclusive privileges to perform CRUD (Create, Read, Update, Delete) operations, allowing them to manage user records, update system data, and ensure the smooth operation of the application. When the admin logs in using their registered phone number and PIN, they gain full access to the entire database, including all user information and transactions.

2.2.2.2 Purpose of USER Table

This table stores information for each user of the GCash mobile application. The phone_number serves as the unique identifier for each user, ensuring secure account access. The first_name and last_name fields enable personalized interactions within the application. The email_address is used for communication and account recovery. The PIN provides an extra layer of security for transactions. This table is important for user authentication and personalization.

2.2.2.3 Purpose of send_transaction Table

This table records all money transfer transactions made by users within the application. Each transaction is assigned a unique send_transaction_ID for easy tracking and reference. The sender_number identifies the user who initiated the transfer, while the receiver_number indicates the recipient. The sender_name and receiver_name fields store the names associated with these phone numbers. The amount field reflects the value of the money transferred in Philippine Peso (PHP). Finally, the transaction_date field documents the date the transfer was completed, following the MM-DD-YYYY format. This table is essential for maintaining a detailed history of all user-initiated fund transfers within the application.

2.2.2.4 Purpose of deposit_transactions Table

The deposit_transactions table serves as the official record of all deposits made by users within the application. It ensures a clear and complete history of every instance where funds are added to a user's wallet. This table meticulously documents important details such as the depositor's name, the amount deposited, and the date of the transaction. By maintaining an accurate and accessible record of deposits, this table enables users to track their account activity.

2.2.2.5 Purpose of withdraw_transactions Table

This table records all withdrawal transactions made by users within the application. It stores essential details such as the unique transaction ID, the withdrawer's numerical identifier, name, the amount withdrawn, and the date of the transaction. This table ensures accurate tracking of withdrawal activities and helps maintain financial records for user accounts.

2.2.2.6 Purpose of WALLET Table

This table records all withdrawal transactions made by users within the application. It stores essential details such as the unique transaction ID, the withdrawer's numerical identifier, name, the amount withdrawn, and the date of the transaction. This table ensures accurate tracking of withdrawal activities and helps maintain financial records for user accounts..

2.2.2.7 Purpose of MONEY Table

This table stores currency-related information used within the application. It includes the currency code as a unique identifier and its corresponding conversion rate relative to a base currency (e.g., USD). This table ensures accurate currency conversions for financial transactions.

2.2.3 Relations

From Table	To Table	Relation
Admin	User	An admin can create, read, update, and delete user accounts.
User	Wallet	A user owns a wallet.
Wallet	Money	A wallet holds money
User	send_transactions	A user can send money to other users.
User	deposit_transactions	A user can deposit money into their wallet.
User	withdraw_transactions	A user can withdraw money from their wallet