PinoyFlix
Database Design Document (DDD)

Version 2.0

Prepared by: Almadrones, Kurt Justine
Calub, John Paul
Dela Paz, Lance Kenneth
Liwanag, Heaven Jameel

Table of Contents

1 Introduction

This document outlines the Database Design Document (DDD) for PinoyFlix, a streaming platform tailored for Filipino audiences, offering a diverse selection of local and international movies, TV shows, and exclusive content. Inspired by platforms like Netflix, PinoyFlix aims to provide an immersive and seamless streaming experience through a well-structured and optimized database system.

1.1 Document Objectives

The objectives of this DDD are:

- To describe the design of databases for PinoyFlix.
- To serve as a blueprint for implementing the database, ensuring it meets the requirements of the platform.
- To provide visibility into the database design for developers, testers, and users.

1.2 Intended Audiences

This document is intended for:

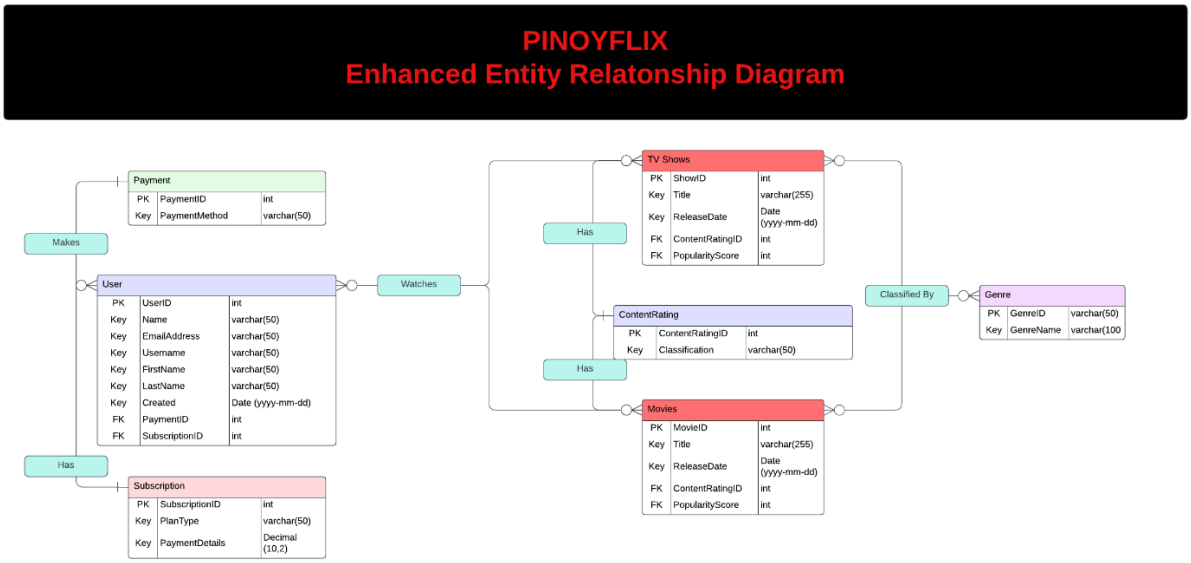
- Panelists and supervisors who will evaluate the quality of the design.
- PinoyFlix developers, including:
    - Designers: To ensure the design aligns with the database structure.
    - Programmers: To implement the database as per the design.
    - Testers: To validate the database against the requirements.

2 Detailed Database Design

This section outlines the database design for PinoyFlix, including both the Entity-Relationship Diagram (ERD) and the Relational (SQLite) Database structure.

2.1 Entity Relationship Diagram (ERD)

Figure 1: Database design



This ERD shows a sample data set and their relationships with other objects. For the propose of the object please refer to the section 2.2.3.

2.1.1 Data dictionary

2.1.1.1 Data dictionary for Element: User Table

Name	Data Type	Constrain	Description
UserID (Primary Key)	INT	AUTO_INCREMENT, PRIMARY KEY	Unique identifier for each user.
Username	VARCHAR(50)	UNIQUE NOT, NULL	**The username of the user.**
FirstName	VARCHAR(50)	NOT NULL	The first name of the user.
LastName	VARCHAR(50)	NOT NULL	The last name of the user.
Email	VARCHAR(100)	UNIQUE NOT, NULL	The email address of the user.
Created	DATETIME	DEFAULT CURRENT_TIMESTAMP	Date and time of creation of account.
PaymentID	INT	FOREIGN KEY	Reference for Payment of user (Cash, Gcash, etc.)
SubscriptionID	INT	FOREIGN KEY	Reference for Subscrption of user (Standard, Premium, etc.)

2.1.1.2 Data dictionary for Element: Subscription Table

Name	Data Type	Constrain	Description
SubscriptionID	INT	AUTO_INCREMENT, PRIMARY KEY	Unique Identifier for each subscription plan.
PlanType	VARCHAR(50)	NOT NULL	The type of subscription plan (e.g., Mobile, Basic Standard, Premium).
Price	DECIMAL(10,2)	NOT NULL	Prices of each Subscriptions.

2.1.1.3 Data dictionary for Element: Payment Method Table

Name	Data Type	Constrain	Description
PaymentID	INT	AUTO_INCREMENT PRIMARY KEY	Unique identifier for each payment method.
PaymentMethod	VARCHAR(50)	NOT NULL	The details of the payment method (e.g., Card, GCash, Maya)

2.1.1.4 Data dictionary for Element: Transaction Table

Name	Data Type	Constrain	Description
TransactionID	INT	PRIMARY KEY	Unique identifier for each transaction.
UserID (Foreign Key)	INT	FOREIGN KEY	References the UserID in the User Table.
PaymentID (Foreign Key)	INT	FOREIGN KEY	References the PaymentID in the Payment Method Table.
SubscriptionID (Foreign Key)	INT	FOREIGN KEY	References the SubscriptionID in the Subscription Table.
Transaction Date	DATETIME	DEFAULT CURRENT_TIMESTAMP	The date when the transaction occurred.

2.1.1.5 Data dictionary for Element: Content Rating Table

Name	Data Type	Constrain	Description
ContentRatingID	INT	PRIMARY KEY	Unique identifier for each content rating.
Classification	VARCHAR(10)	NOT NULL UNIQUE	The classification of the content (e.g., PG, PG-13, R-16, R-18).

2.1.1.6 Data dictionary for Element: TV Shows Table

Name	Data Type	Constrain	Description
ShowID (Primary Key)	INT	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for each TV show.
Title	VARCHAR(255)	NOT NULL	The title of the TV show.
ReleaseDate	DATE		The release date of the TV show.
ContentRatingID	INT	FOREIGN KEY	References the content rating classification of the TV show.

PopularityScore	INT		The popularity score of the TV show.

2.1.1.7 Data dictionary for Element: Movies Table

Name	Data Type	Constrain	Description
MovieID (Primary Key)	INT	PRIMARY KEY, AUTO_INCREMENT	Unique identifier for each movie.
Title	VARCHAR(255)	NOT NULL	The title of the movie.
ReleaseDate	DATE	UNIQUE	The release date of the movie.
ContentRatingID	INT	FOREIGN KEY	The content rating classification of the movie.
PopularityScore	INT		The popularity score of the movie.

2.1.1.8 Data dictionary for Element: Genre Table

Name	Data Type	Constrain	Description
GenreID	INT	**AUTO_INCREMENT PRIMARY KEY**	Unique identifier for each genre.
GenreName	VARCHAR(100)		The name of the genre (e.g., Action, Comedy, Drama).

2.1.1.9 Data dictionary for Element: Genre Shows Table

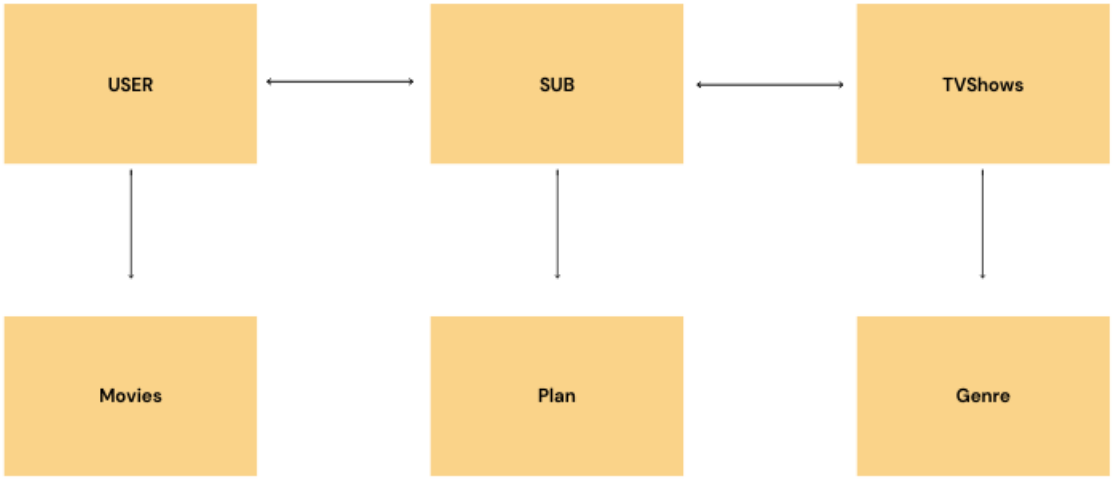
Name	Data Type	Constrain	Description
ShowID	Integer	FOREIGN KEY	References the ShowID in the TV Shows Table.
GenreID (Foreign Key)	Integer	FOREIGN KEY	References the GenreID in the Genre Table.

2.1.1.10 Data dictionary for Element: Genre Movies Table

Name	Data Type	Constrain	Description
MovieID (Foreign Key)	Integer	FOREIGN KEY	References the MovieID in the Movies Table.
GenreID (Foreign Key)	Integer	FOREIGN KEY	References the GenreID in the Genre Table.

2.2 Database Design (Relational database)

2.2.1 Conceptual diagram



2.2.2 Description

This diagram is all about how Pinoyflix works behind the scenes it's like the blueprint that keeps everything running smoothly. At the heart of it are the users, each with their own account where we store basics like their name, email, and a unique UserID. Users pick a subscription plan like basic or premium which decides what kind of content they can watch. Over in the TVShows and Movies sections, we keep track of all the content details: titles, release dates, genres, and ratings, with each show and movie having its own unique ID (ShowID or MovieID) to avoid any mix-ups. The Subscription section handles the different plans and payment details, making sure users get access to the right content based on what they've signed up for. Everything's connected in a way that just works users log in, browse their favorite shows or movies, and enjoy without a hitch. In short, this diagram is the foundation that keeps Pinoyflix organized and ready to deliver a great experience for everyone.

2.2.3 Purpose of Tables

2.2.3.1 Purpose of TVShows Table

This table stores information about the TV shows available on Pinoyflix. Each show has a unique ShowID, along with details like the title, release date, genre, and rating. This table helps organize and manage the TV show content that users can access based on their subscription plans.

2.2.3.2 Purpose of Movies Table

This table stores information about the movies available on Pinoyflix. Each movie has a unique MovieID, along with details like the title, release date, and rating. This table ensures that movie content is properly organized and accessible to users based on their subscription type.

2.2.3.3 Purpose of User Table

This table stores the details of users registered on Pinoyflix. Each user has a unique UserID, along with their name and email address. The email address serves as a key identifier for the user. This table also links users to their subscription plans, ensuring they have access to the appropriate content.

2.2.3.4 Purpose of Subscription Table

This table manages the subscription plans available on Pinoyflix. Each subscription has a unique SubscriptionID, along with details like the plan type (e.g., basic, premium) and payment information. This table ensures that users are granted access to content based on their chosen subscription plan

2.2.4 Relations

From Table	To Table	Relation
User	Subscription	A user has one subscription plan.
Subscription	User	A subscription plan deterimenes what subscription type of account.
Payment	User	A payment method deterimenes what payment type of account.
User	TVShows	A user can see TV Shows in the table
User	Movies	A user can see Movies in the table
User	Payment	A user will choose a Payment Method
Genre	TVShows	A Tvshows can have multiple Genres.
Genre	Movies	A Movie can have multiple Genres.