

# Partial Project Definition

Diego Alejandro Garzón Rodríguez - 20222020009 Daniel Josué Rozo Rodríguez - 20222020006

**Francisco José de Caldas District University**

**Databases Foundation - Teacher:** Carlos Andres Sierra Virguez

**Abstract**—The main objective of the project is to develop a system like Steam, a digital video game distribution platform, focusing especially on the design and implementation of a relational database that supports all system functionalities. The database will serve as the core of the system, storing crucial information such as registered user details, available game details, purchase transactions, personal game libraries of each user, game reviews, and ratings, among other relevant data. The structure of the database will be carefully designed to ensure integrity, consistency, and efficiency in data storage and retrieval. Concepts learned in class about relational database design, normalization, primary and foreign keys will be utilized, as well as techniques to optimize queries and ensure system scalability.

**Index Terms**—Steam, digital platform, store, games, system

## I. INTRODUCTION

IN this work we will see a database design focused on replicating in the simplest but most efficient way the application “steam”, known worldwide for being one of the largest online video game stores. We will focus on its basic features, those that make it a useful application for all gamers. We will define the entities, relationships and schemes needed to make the designed database understandable, flexible and above all effective.

To give context, we would like to comment a little on the history of the application; Steam is a digital game distribution platform developed by Valve Corporation. It was released in September 2003 and allows users to purchase, download and play computer games, as well as software applications and multimedia content. Steam also offers video game streaming services, such as Steam Remote Play Together, which allows users to play games online with friends and share their screens in real time. In general terms over the years to date Steam is a virtual store that allows users to buy and download games digitally.

Among the main features that must have Steam and any digital store that intends to replicate it we find:

- A clear product catalogue
- Efficient and secure data management
- A shopping cart or system

And it's these kinds of things that will give us a road map to start designing the database system that the application will have to materialize later in the next phase of the project.

### A. 1. Description of the project

The project consists of designing a database that simulates an online video game store, specifically looking for this to be a replica of the famous desktop application “Steam”. The idea is to provide, in a simple and as objective way as possible, a design that adapts to the proposed needs, maintains a stable and large volume of data (as it usually is), but at the same time simplifies the entire system in the best way possible. We are two members in charge of carrying out the project.

Steam by default is a desktop application, although it is also designed to display the catalog and features in a web browser (web application), but we will not focus much on that. It has an initial interface that receives the data of “username” and the respective password. The business model of the application is simple, sales, since it is a virtual store, handles the purchase of video games, live broadcasts of them by other users (to promote the products), a personal library for each user and a “community” section that at the same time includes its own “mini market”. The application also has a section of free games where, as in the previous section, we can purchase and install them from the application.

Inside its interface it manages a menu with six options and a keyboard search bar. It divides the games into their different categories, prices and other features. It's a complete shop.

The tools we will use (this when we pass the planning and design stage) are:

- MySQL
- Visual Studio (for Python)
- Draw.IO
- Relational Algebra
- Among others to be added as development requires

It is important to clarify that the system itself has an immense magnitude and scope, given that it is an application of global character, the idea is to replicate it but for practical purposes soon there will be functions with which application counts that come to be discarded in the final model. Not because we don't want to replicate it well, but because we will take more account of the most vital and important features that make the app one of the most used online stores today.

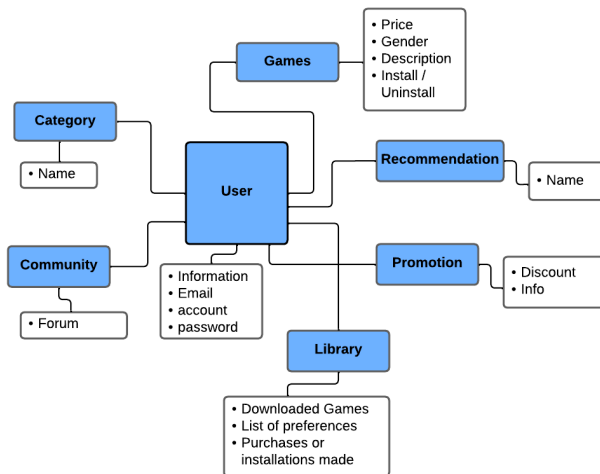
### B. 2. User stories

- As a user I want to know which games have discounts or special prices, in case they are presented

- As a user I want to see the catalogs of the games organized by categories, to be able to search the games you want in an orderly way
- As a user I want you to recommend me games, to try new titles and expand my repertoire
- As a user I want to receive notifications of new video game releases when they are released
- As a user I want a library, where I can see all the games I have purchased or downloaded
- As a user I want to be able to install and uninstall my games easily from the same application
- As a user I want a community of more users within the same application, to be able to interact about the games and know more about them
- As a user I want to be shown all the features and details of a game when accessing the download page of the same

### C. 3. The conceptual model of the project

#### • Step 0. Define components:



#### • Step 1. Define entities:

- E1. User
- E2. Games
- E3. Library
- E4. Community
- E5. Category
- E6. Promotion
- E7. Recommendation

#### • Step 2. Define attributes per entity:

- E1: UserID, name, email, account, password
- E2: GamesID, name, price, description, install / uninstall option
- E3: LibraryID, downloaded games, list of preference, purchases or installations made
- E4: CommunityID, forum
- E5: CategoryID, name
- E6: PromotionID, discount, info
- E7: RecommendationID, name

#### Step 3. Define Relationships:

	E1	E2	E3	E4	E5	E6	E7
E1							
E2							
E3							
E4							
E5							
E6							
E7							

Fig. 1. Step 3

#### Step 4. Define relationships type:

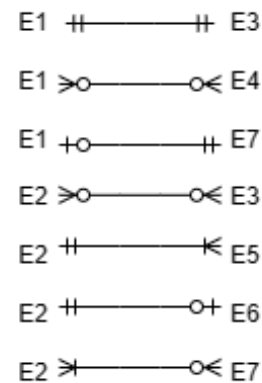


Fig. 2. Step 4

#### Step 5. First Entity-Relationship Draw:

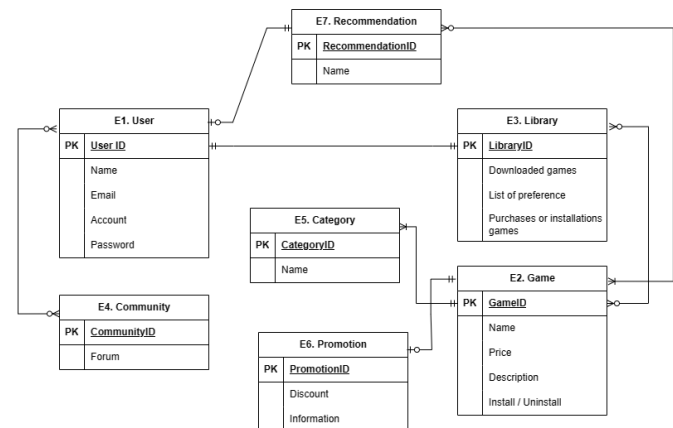


Fig. 3. Step 5

#### Step 6-8. First Split Many-to-Many Relationships and the Second Entity-Relationship Draw:

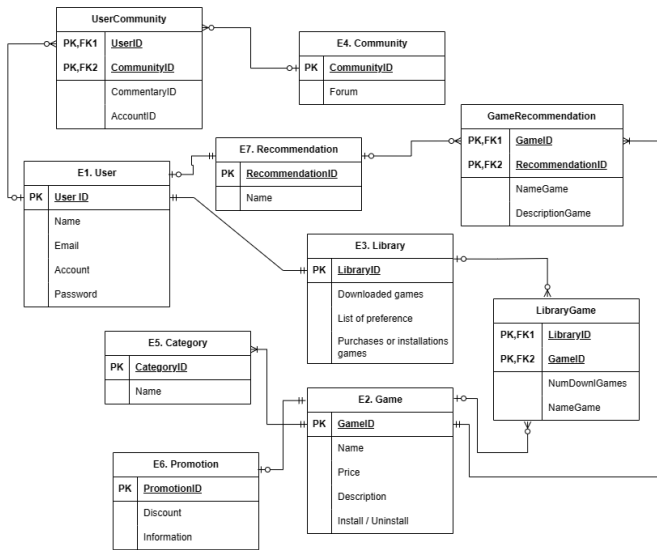


Fig. 4. Step 8

### Step 9-10. Get Data-Structure E-R M and Define Constraints and Properties of Data:

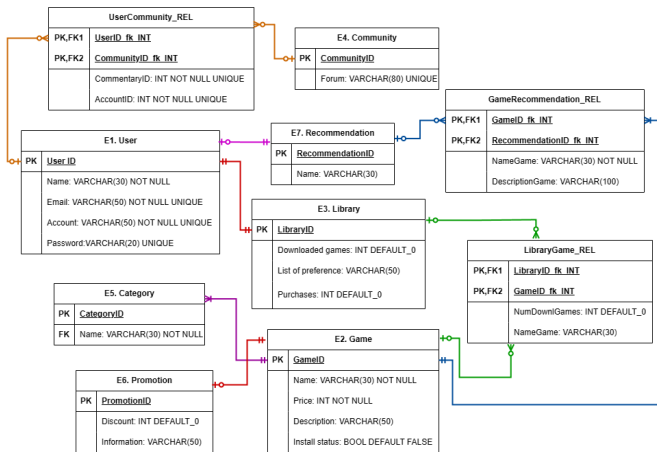


Fig. 5. Step 10

### D. 4. All the relational algebra queries you think will be useful in your project

a) The name of the games that have discounts or special prices:

- `name( GameID=game_promotions_id(Game))`
- `( game_promotions_id( game_id(Promotion)))`

b) The name of the games that are for "Sports" category:

- `name( category_id=sports_id(Game))`
- `( sports_id( CategoryID( name="Sports"(Category)))`

c) The name of the games that are in the library of the user "Peipito123"

- `name( GameID=pepito123_games_id(Game))`

- `( pepito123_games_id( game_id( Li-  
braryID=pepito123_libraty_id(Library)))`
- `( pepito123_library_id( libriry_id(  
name="Pepito123"(User)))`

### E. 5. Search for data on the web to figure out free data sources to populate any of your entities

**Mockaroo:** Mockaroo is a versatile online tool designed to assist users in generating realistic test data for various purposes. It offers a user-friendly interface that allows individuals to create customized datasets tailored to their specific needs.

The tool provides a wide range of data types, including names, addresses, phone numbers, email addresses, dates, and more. Users can define the format, constraints, and other characteristics of each data field, ensuring that the generated data aligns with their testing requirements.

One of the key features of Mockaroo is its flexibility. Users can specify the number of rows of data they need, making it suitable for generating small sample datasets for testing individual components or large datasets for comprehensive system testing.

Additionally, Mockaroo supports the generation of data in various formats, including CSV, JSON, SQL, and Excel. This versatility enables users to seamlessly integrate the generated data into their applications, databases, or testing environments.

Mockaroo is widely used by developers, testers, and data analysts to populate databases during development and testing phases, create sample datasets for demonstrations and presentations, and simulate real-world scenarios for analysis and validation purposes.

## II. CONCLUSION

- Replicating the database system of an application the size of Steam that has a global reach and over 100 million users can become complex if we don't focus on the basic functionalities and the basic data they need to operate. The system should be scalable as much as it can look more like the original system, but if not, the essence of what makes a digital store should be kept.

- Having clear user histories is vital for the correct design of a database system, as the project was carried out, we had to change and reorder the entities to work because they did not allow a correct analysis of the application and we came to significant conceptual flaws. Therefore, the definition of requirements and entities is, in our end, the most important step of the design.

- Less is more, the better the basic functions of the program are performed, and the faster and more efficient the data management, the better.

## REFERENCES

- [1] Cabácas, T. (2018). "15 years of Steam, the platform that changed the way we play". Tomado y adaptado de: 15 años de Steam, la plataforma que cambió nuestra forma de jugar (muycomputer.com)

[2] Kiero. (2022). “12 Characteristics of a Successful Online Store”. Tomado y adaptado de: 12 características de una tienda online exitosa - Plataforma Ecommerce Kiero: Menú digital — Tienda Online Sin Comisiones

[3] Mockaroo. (2022). “Mockaroo”. Tomado y adaptado de: Mockaroo - Random Data Generator and API Mocking Tool — JSON / CSV / SQL / Excel

[4] RetabloWeb. (2021). “¿What is an Online Store? Characteristics, Elements and Types.”. Tomado y adaptado de: ¿Qué es una Tienda Online? Características, Elementos y Tipos. - RetabloWeb