Trinity College Dublin

The University of Dublin

INFORMATION MANAGEMENT FOR ENGINEERING

CSU44D01-202223

PROJECT ASTERIX – A Video Game Distribution System

Name: Mudit Garg

TCD ID: 21355125

E-Mail ID: gargmu@tcd.ie

Mobile: +353892050748

INTRODUCTION

The application chosen for the purpose of representing information in the form of an ER model is of a video game distribution platform with similarities taken from real world examples like **Steam** by Value, **Origin** by Electronic Arts or **Epic Games**, etc.

During the COVID period, when the whole world was confined to their homes, the gaming industry saw a huge boom. With the huge catalogue of a variety of games and the inaccessibility of physical shops, the importance of maintaining and providing a digital forefront for the distribution of games started to gain traction.

This following report contains most of the details for how I would go about implementing such a digital platform's database to manage the huge amount of data that accompanies with a popular video game distribution platform such as data about its *Users*, the hosted *Games*, the various *Genres* the hosted games are in, the *Languages* the platform supports, the various *Offers* the website currently holds for its users and the various *News* related to the various Publishers and the Games on its platform.

ENTITIES AND THEIR ATTRIBUTES

- *Users* – This table contains the list of all the users and their details, that are currently registered with the platform.

  Columns: User_ID (Primary Key), First Name, Last Name, Email, Password, Language

  Table Creation:

```
CREATE TABLE `users` (
  `User_ID` int NOT NULL AUTO_INCREMENT,
  `First Name` varchar(30) NOT NULL,
  `Last Name` varchar(45) NOT NULL,
  `Email` varchar(20) NOT NULL,
  `Password` varchar(20) NOT NULL,
  `Language` varchar(10) NOT NULL DEFAULT 'English',
  PRIMARY KEY (`User_ID`),
  UNIQUE KEY `User_ID_UNIQUE` (`User_ID`),
  UNIQUE KEY `Email_UNIQUE` (`Email`)
) COMMENT='List of all the users and their details in the database'
```

  Table Population:

```
INSERT INTO `server_database`.`users` (`User_ID`, `First Name`, `Last Name`, `Email`, `Password`, `Language`) VALUES ('1', 'Mudit', 'Garg', 'gargmu@tcd.ie', 'FunnyNinja23', 'English');
INSERT INTO `server_database`.`users` (`User_ID`, `First Name`, `Last Name`, `Email`, `Password`, `Language`) VALUES ('2', 'Rahul', 'Garg', 'gargr@tcd.ie', '3flex@', 'Hindi');
INSERT INTO `server_database`.`users` (`User_ID`, `First Name`, `Last Name`, `Email`, `Password`, `Language`) VALUES ('3', 'Yashraj', 'Saluja', 'yashraj@gmail.com', 'maakaladla134', 'Punjabi');
```

```
INSERT INTO `server_database`.`users` (`User_ID`, `First Name`, `Last Name`, `Email`, `Password`, `Language`) VALUES ('4', 'Angad', 'Singh Garcha', 'garchaa@outlook.com', 'ikyktk69', 'French');
INSERT INTO `server_database`.`users` (`User_ID`, `First Name`, `Last Name`, `Email`, `Password`, `Language`) VALUES ('5', 'Ritwik', 'Kumar', 'kumarritwik@aol.com', 'coolDude13', 'Tamil');
INSERT INTO `server_database`.`users` (`User_ID`, `First Name`, `Last Name`, `Email`, `Password`, `Language`) VALUES ('6', 'Ananya', 'Garg', 'papaKipari@yahoo.com', 'cakemurder01', 'Hindi');
INSERT INTO `server_database`.`users` (`User_ID`, `First Name`, `Last Name`, `Email`, `Password`, `Language`) VALUES ('7', 'Mathew', 'McHart', 'mathewmc@aol.com', 'asdfmovie%', 'English');
```

Final Table:

User_ID	First Name	Last Name	Email	Password	Language
1	Mudit	Garg	gargmu@tcd.ie	FunnyNinja23	English
2	Rahul	Garg	gargr@tcd.ie	3flex@	Hindi
3	Yashraj	Saluja	yashraj@gmail.com	maakaladla134	Punjabi
4	Angad	Singh Garcha	garchaa@outlook.com	ikyktk69	French
5	Ritwik	Kumar	kumarritwik@aol.com	coolDude13	Tamil
6	Ananya	Garg	papaKipari@yahoo.com	cakemurder01	Hindi
7	Mathew	McHart	mathewmc@aol.com	asdfmovie%	English
NULL	NULL	NULL	NULL	NULL	NULL

- *Games* – This table stores all the various games that are registered with the platform by the publishers for distribution among the users.

  Columns: Game_ID (Primary Key), Name, Price, Release Date

  Table Creation:

```
CREATE TABLE `games` (
  `Game_ID` int NOT NULL AUTO_INCREMENT,
  `Name` varchar(60) NOT NULL,
  `Price` float NOT NULL DEFAULT '0',
  `Release Date` datetime NOT NULL,
  PRIMARY KEY (`Game_ID`),
  UNIQUE KEY `Game_ID_UNIQUE` (`Game_ID`)
) COMMENT='List of all the games present in the database'
```

  Table Population:

```
INSERT INTO `server_database`.`games` (`Game_ID`, `Name`, `Price`, `Release Date`) VALUES ('1', 'Uncharted', '23.99', '2015-10-21')
INSERT INTO `server_database`.`games` (`Game_ID`, `Name`, `Price`, `Release Date`) VALUES ('2', 'The Spiderman', '14.99', '2021-11-17')
INSERT INTO `server_database`.`games` (`Game_ID`, `Name`, `Price`, `Release Date`) VALUES ('3', 'GTA V', '8.99', '2013-09-10')
```

INSERT INTO `server_database`.`games` (`Game_ID`, `Name`, `Price`, `Release Date`) VALUES ('4', 'Red Dead Redemption', '18.99', '2018-06-16')
INSERT INTO `server_database`.`games` (`Game_ID`, `Name`, `Price`, `Release Date`) VALUES ('5', 'FIFA 19', '20.99', '2019-04-12')
INSERT INTO `server_database`.`games` (`Game_ID`, `Name`, `Price`, `Release Date`) VALUES ('6', 'Cyberpunk 2077', '15.99', '2020-11-30')
INSERT INTO `server_database`.`games` (`Game_ID`, `Name`, `Price`, `Release Date`) VALUES ('7', 'Spiderman: Miles Morales', '28.99', '2022-11-17')
INSERT INTO `server_database`.`games` (`Game_ID`, `Name`, `Price`, `Release Date`) VALUES ('8', 'Witcher 3', '32.99', '2018-03-25')
INSERT INTO `server_database`.`games` (`Game_ID`, `Name`, `Price`, `Release Date`) VALUES ('9', 'GTA: San Andreas', '0.00', '2009-08-14')

Final Table:

Game_ID	Name	Price	Release Date
1	Uncharted	23.99	2015-10-21 …
2	The Spiderman	14.99	2021-11-17 …
3	GTA V	8.99	2013-09-10 …
4	Red Dead Redemption 2	18.99	2018-06-16 …
5	FIFA 19	20.99	2019-04-12 …
6	Cyberpunk 2077	15.99	2020-11-30 …
7	Spiderman: Miles Morales	28.99	2022-11-17 …
8	Witcher 3	32.99	2018-03-25 …
9	GTA: San Andreas	0	2009-08-14 …
NULL	NULL	NULL	NULL

- *Library* – This entity is responsible for encapsulating all the games that every user has ever purchased on the platform.

  Columns: Library Item ID (Primary Key), library_id (Foreign Key) (User ID from Users table), game_ids (Foreign Key) (Game ID from Games table)

  Table Creation:

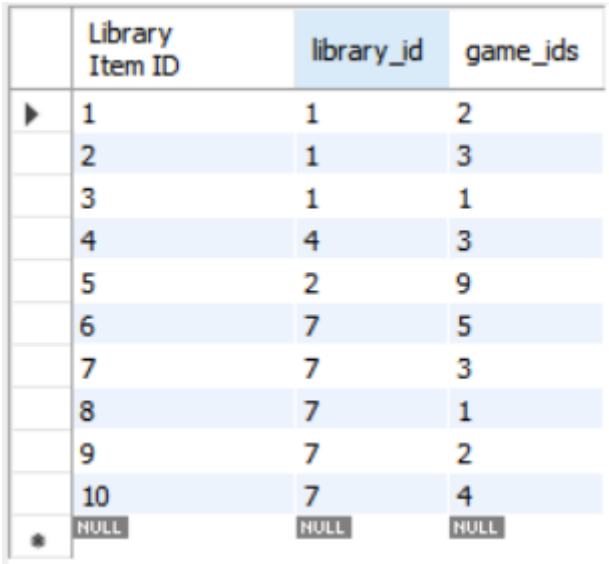
```
CREATE TABLE `user_libraries` (
  `Library Item ID` int NOT NULL AUTO_INCREMENT,
  `library_id` int NOT NULL,
  `game_ids` int NOT NULL,
  PRIMARY KEY (`Library Item ID`),
  KEY `Library_ID_idx` (`library_id`),
  KEY `Game_ID_idx` (`game_ids`),
  CONSTRAINT `game_ids_library` FOREIGN KEY (`game_ids`) REFERENCES `games` (`Game_ID`),
```

CONSTRAINT `library_id` FOREIGN KEY (`library_id`) REFERENCES `users` (`User_ID`)
) COMMENT='Table that stores all the library items of all the users in the database by using the User_ID from the Users table and the Game_ID from the Games table'

Table Population:

```
INSERT INTO `server_database`.`user_libraries` (`Library Item ID`, `library_id`, `game_ids`) VALUES ('1', '1', '2')
INSERT INTO `server_database`.`user_libraries` (`Library Item ID`, `library_id`, `game_ids`) VALUES ('2', '1', '3')
INSERT INTO `server_database`.`user_libraries` (`Library Item ID`, `library_id`, `game_ids`) VALUES ('3', '1', '1')
INSERT INTO `server_database`.`user_libraries` (`Library Item ID`, `library_id`, `game_ids`) VALUES ('4', '4', '3')
INSERT INTO `server_database`.`user_libraries` (`Library Item ID`, `library_id`, `game_ids`) VALUES ('5', '2', '9')
INSERT INTO `server_database`.`user_libraries` (`Library Item ID`, `library_id`, `game_ids`) VALUES ('6', '7', '5')
INSERT INTO `server_database`.`user_libraries` (`Library Item ID`, `library_id`, `game_ids`) VALUES ('7', '7', '3')
INSERT INTO `server_database`.`user_libraries` (`Library Item ID`, `library_id`, `game_ids`) VALUES ('8', '7', '1')
INSERT INTO `server_database`.`user_libraries` (`Library Item ID`, `library_id`, `game_ids`) VALUES ('9', '7', '2')
INSERT INTO `server_database`.`user_libraries` (`Library Item ID`, `library_id`, `game_ids`) VALUES ('10', '7', '4')
```

Final Table:



Library Item ID	library_id	game_ids
1	1	2
2	1	3
3	1	1
4	4	3
5	2	9
6	7	5
7	7	3
8	7	1
9	7	2
10	7	4
NULL	NULL	NULL

- *Wishlist* – All the games that every user has added to their wish list is stored in this table.

Columns: Wishlist Item ID (Primary Key), wishlist_id (Foreign Key) (User ID from Users table), game_ids (Foreign Key) (Game ID from Games table)

Table Creation:

```
CREATE TABLE `user_wishlists` (
  `Wishlist Item ID` int NOT NULL AUTO_INCREMENT,
  `wishlist_id` int NOT NULL,
  `game_ids` int NOT NULL,
  PRIMARY KEY (`Wishlist Item ID`),
  KEY `User_ID_idx` (`wishlist_id`),
  KEY `Game_ID_idx` (`game_ids`),
  CONSTRAINT `game_ids_wishlist` FOREIGN KEY (`game_ids`) REFERENCES `games` (`Game_ID`),
  CONSTRAINT `Wishlist_ID` FOREIGN KEY (`wishlist_id`) REFERENCES `users` (`User_ID`)
) COMMENT='Table that stores all the wishlists of all the users in the database by using the User_ID from the Users table and the Game_ID from the Games table'
```

Table Population:

```
INSERT INTO `server_database`.`user_wishlists` (`Wishlist Item ID`, `wishlist_id`, `game_ids`) VALUES ('1', '1', '7')
INSERT INTO `server_database`.`user_wishlists` (`Wishlist Item ID`, `wishlist_id`, `game_ids`) VALUES ('2', '1', '6')
INSERT INTO `server_database`.`user_wishlists` (`Wishlist Item ID`, `wishlist_id`, `game_ids`) VALUES ('3', '2', '3')
INSERT INTO `server_database`.`user_wishlists` (`Wishlist Item ID`, `wishlist_id`, `game_ids`) VALUES ('4', '2', '8')
INSERT INTO `server_database`.`user_wishlists` (`Wishlist Item ID`, `wishlist_id`, `game_ids`) VALUES ('5', '6', '3')
INSERT INTO `server_database`.`user_wishlists` (`Wishlist Item ID`, `wishlist_id`, `game_ids`) VALUES ('6', '5', '5')
INSERT INTO `server_database`.`user_wishlists` (`Wishlist Item ID`, `wishlist_id`, `game_ids`) VALUES ('7', '4', '5')
INSERT INTO `server_database`.`user_wishlists` (`Wishlist Item ID`, `wishlist_id`, `game_ids`) VALUES ('8', '4', '8')
INSERT INTO `server_database`.`user_wishlists` (`Wishlist Item ID`, `wishlist_id`, `game_ids`) VALUES ('9', '3', '5')
INSERT INTO `server_database`.`user_wishlists` (`Wishlist Item ID`, `wishlist_id`, `game_ids`) VALUES ('10', '3', '2')
```

Final Table:

Wishlist Item ID	wishlist_id	game_ids
1	1	7
2	1	6
3	2	3
4	2	8
5	6	3
6	5	5
7	4	5
8	4	8
9	3	5
10	3	2
NULL	NULL	NULL

- *Cart* – Whenever a user decides to buy one or more games, he/she adds those games into their cart. This entity is the one which stores all the games of all the users which every user has added into their cart.

  Columns: Cart Item ID (Primary Key), cart_id (Foreign Key) (User ID from Users table), game_ids (Foreign Key) (Game ID from Games table)

  Table Creation:

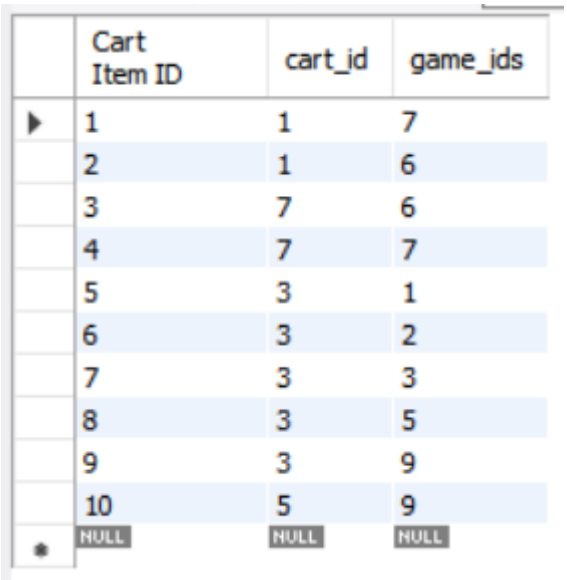
  CREATE TABLE `user_carts` (
  `Cart Item ID` int NOT NULL AUTO_INCREMENT,
  `cart_id` int NOT NULL,
  `game_ids` int NOT NULL,
  PRIMARY KEY (`Cart Item ID`),
  KEY `User_ID_idx` (`cart_id`),
  KEY `Game_ID_idx` (`game_ids`),
  CONSTRAINT `cart_id` FOREIGN KEY (`cart_id`) REFERENCES `users` (`User_ID`),
  CONSTRAINT `game_ids_cart` FOREIGN KEY (`game_ids`) REFERENCES `games` (`Game_ID`)
) COMMENT='Table that stores all the cart items of all the users in the database by using the User_ID from the Users table and the Game_ID from the Games table'

  Table Population:

  INSERT INTO `server_database`.`user_carts` (`Cart Item ID`, `cart_id`, `game_ids`) VALUES ('1', '1', '7')
  INSERT INTO `server_database`.`user_carts` (`Cart Item ID`, `cart_id`, `game_ids`) VALUES ('2', '1', '6')
  INSERT INTO `server_database`.`user_carts` (`Cart Item ID`, `cart_id`, `game_ids`) VALUES ('3', '7', '6')

```
INSERT INTO `server_database`.`user_carts` (`Cart Item ID`, `cart_id`, `game_ids`) VALUES ('4', '7', '7')
INSERT INTO `server_database`.`user_carts` (`Cart Item ID`, `cart_id`, `game_ids`) VALUES ('5', '3', '1')
INSERT INTO `server_database`.`user_carts` (`Cart Item ID`, `cart_id`, `game_ids`) VALUES ('6', '3', '2')
INSERT INTO `server_database`.`user_carts` (`Cart Item ID`, `cart_id`, `game_ids`) VALUES ('7', '3', '3')
INSERT INTO `server_database`.`user_carts` (`Cart Item ID`, `cart_id`, `game_ids`) VALUES ('8', '3', '5')
INSERT INTO `server_database`.`user_carts` (`Cart Item ID`, `cart_id`, `game_ids`) VALUES ('9', '3', '9')
INSERT INTO `server_database`.`user_carts` (`Cart Item ID`, `cart_id`, `game_ids`) VALUES ('10', '5', '9')
```

Final Table:



Cart Item ID	cart_id	game_ids
1	1	7
2	1	6
3	7	6
4	7	7
5	3	1
6	3	2
7	3	3
8	3	5
9	3	9
10	5	9
NULL	NULL	NULL

- *Platforms* – This platform also has the feature that enables the publishers to distribute their games to various platforms. Some of the examples that are considered here are PC (Windows, Mac, Linux, Chrome OS), Xbox and PlayStation. This table is the one which contains the various platforms on which every game is published and playable by the user.

  Columns: Platform Item ID (Primary Key), Game_ID (Foreign Key) (Game ID from Games table), Platform ID, Platform Name

  Table Creation:

```
CREATE TABLE `platforms` (
  `Platform Item ID` int NOT NULL AUTO_INCREMENT,
  `Game_ID` int NOT NULL,
  `Platform_ID` int NOT NULL,
```

```
  `Platform Name` varchar(45) NOT NULL,
  PRIMARY KEY (`Platform Item ID`),
  KEY `game_ids_platform_idx` (`Game_ID`),
  CONSTRAINT `game_ids_platform` FOREIGN KEY (`Game_ID`) REFERENCES
`games` (`Game_ID`)
) COMMENT='List of all the platforms that all the games in the database are released
for'
```

Table Population:

```
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('1', '1', '1', 'PC')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('2', '1', '2', 'Xbox')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('3', '1', '3', 'Playstation')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('4', '2', '1', 'PC')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('5', '2', '2', 'Xbox')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('6', '2', '3', 'Playstation')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('7', '3', '1', 'PC')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('8', '4', '2', 'Xbox')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('9', '4', '3', 'Playstation')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('10', '5', '1', 'PC')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('11', '6', '1', 'PC')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('12', '7', '1', 'PC')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('13', '7', '3', 'Playstation')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('14', '8', '1', 'PC')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('15', '8', '2', 'Xbox')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('16', '8', '3', 'Playstation')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('17', '9', '1', 'PC')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('18', '9', '2', 'Xbox')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('19', '9', '3', 'Playstation')
INSERT INTO `server_database`.`platforms` (`Platform Item ID`, `Game_ID`, `Platform_ID`, `Platform
Name`) VALUES ('20', '9', '4', 'Android')
```

Final Table:

	Platform Item ID	Game_ID	Platform_ID	Platform Name
▶	1	1	1	PC
	2	1	2	Xbox
	3	1	3	Playstation
	4	2	1	PC
	5	2	2	Xbox
	6	2	3	Playstation
	7	3	1	PC
	8	4	2	Xbox
	9	4	3	Playstation
	10	5	1	PC
	11	6	1	PC
	12	7	1	PC
	13	7	3	Playstation
	14	8	1	PC
	15	8	2	Xbox
	16	8	3	Playstation
	17	9	1	PC
	18	9	2	Xbox
	19	9	3	Playstation
	20	9	4	Android
●	NULL	NULL	NULL	NULL

- *Publishers* – This table stores the names of the various publishers that have their games on the website.

  Columns: Publisher Item ID (Primary Key), Publisher ID, Publisher Name, Game ID (Foreign Key)

  Table Creation:

```
CREATE TABLE `publishers` (
  `Publisher Item ID` int NOT NULL AUTO_INCREMENT,
  `Publisher_ID` int NOT NULL,
  `Publisher Name` varchar(60) NOT NULL,
  `Game_ID` int NOT NULL,
  PRIMARY KEY (`Publisher Item ID`),
  KEY `game_id_publisher_idx` (`Game_ID`),
  CONSTRAINT `game_id_publisher` FOREIGN KEY (`Game_ID`) REFERENCES
`games` (`Game_ID`)
) COMMENT='Table that contains all the games that each of the games in the database
belong to'
```

Table Population:

```
INSERT INTO `server_database`.`publishers` (`Publisher Item ID`, `Publisher_ID`, `Publisher Name`, `Game_ID`) VALUES ('1', '1', 'Sony Interactive Entertainment', '1')
INSERT INTO `server_database`.`publishers` (`Publisher Item ID`, `Publisher_ID`, `Publisher Name`, `Game_ID`) VALUES ('2', '1', 'Sony Interactive Entertainment', '2')
INSERT INTO `server_database`.`publishers` (`Publisher Item ID`, `Publisher_ID`, `Publisher Name`, `Game_ID`) VALUES ('3', '1', 'Sony Interactive Entertainment', '7')
INSERT INTO `server_database`.`publishers` (`Publisher Item ID`, `Publisher_ID`, `Publisher Name`, `Game_ID`) VALUES ('4', '2', 'Rockstar Games', '3')
INSERT INTO `server_database`.`publishers` (`Publisher Item ID`, `Publisher_ID`, `Publisher Name`, `Game_ID`) VALUES ('5', '2', 'Rockstar Games', '9')
INSERT INTO `server_database`.`publishers` (`Publisher Item ID`, `Publisher_ID`, `Publisher Name`, `Game_ID`) VALUES ('6', '2', 'Rockstar Games', '4')
INSERT INTO `server_database`.`publishers` (`Publisher Item ID`, `Publisher_ID`, `Publisher Name`, `Game_ID`) VALUES ('7', '3', 'Electronic Arts', '5')
INSERT INTO `server_database`.`publishers` (`Publisher Item ID`, `Publisher_ID`, `Publisher Name`, `Game_ID`) VALUES ('8', '4', 'CD Projekt Red', '6')
INSERT INTO `server_database`.`publishers` (`Publisher Item ID`, `Publisher_ID`, `Publisher Name`, `Game_ID`) VALUES ('9', '4', 'CD Projekt Red', '8')
```

Final Table:

Publisher Item ID	Publisher_ID	Publisher Name	Game_ID
1	1	Sony Interactive Entertainment	1
2	1	Sony Interactive Entertainment	2
3	1	Sony Interactive Entertainment	7
4	2	Rockstar Games	3
5	2	Rockstar Games	9
6	2	Rockstar Games	4
7	3	Electronic Arts	5
8	4	CD Projekt Red	6
9	4	CD Projekt Red	8
NULL	NULL	NULL	NULL

- *Genres* – Every game also has a detail about which genre the game belongs to. This table is responsible for storing that information

  Columns: Genre Item ID (Primary Key), Genre ID, Genre Name, Game ID (Foreign Key)

  Table Creation:

```
CREATE TABLE `genres` (
  `Genre Item ID` int NOT NULL AUTO_INCREMENT,
  `Genre_ID` int NOT NULL,
```

```
  `Genre Name` varchar(45) NOT NULL,
  `Game_ID` int NOT NULL,
  PRIMARY KEY (`Genre Item ID`),
  KEY `game_id_genre_idx` (`Game_ID`),
  CONSTRAINT `game_id_genre` FOREIGN KEY (`Game_ID`) REFERENCES `games` (`Game_ID`)
) COMMENT='Table that contains all the genres that each of the games in the database belong to'
```

Table Population:

```
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('1', '1', 'Action', '1')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('2', '2', 'Adventure', '1')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('3', '1', 'Action', '2')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('4', '2', 'Adventure', '2')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('5', '1', 'Action', '3')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('6', '2', 'Adventure', '3')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('7', '1', 'Action', '4')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('8', '2', 'Adventure', '4')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('9', '3', 'Unique', '4')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('10', '4', 'Sports', '5')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('11', '5', 'RPG', '6')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('12', '1', 'Action', '7')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('13', '2', 'Adventure', '7')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('14', '3', 'Unique', '8')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('15', '5', 'RPG', '8')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('16', '1', 'Action', '9')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('17', '2', 'Adventure', '9')
INSERT INTO `server_database`.`genres` (`Genre Item ID`, `Genre_ID`, `Genre Name`, `Game_ID`) VALUES ('18', '5', 'RPG', '9')
```

Final Table:

	Genre Item ID	Genre_ID	Genre Name	Game_ID
▶	1	1	Action	1
	2	2	Adventure	1
	3	1	Action	2
	4	2	Adventure	2
	5	1	Action	3
	6	2	Adventure	3
	7	1	Action	4
	8	2	Adventure	4
	9	3	Unique	4
	10	4	Sports	5
	11	5	RPG	6
	12	1	Action	7
	13	2	Adventure	7
	14	3	Unique	8
	15	5	RPG	8
	16	1	Action	9
	17	2	Adventure	9
	18	5	RPG	9
*	NULL	NULL	NULL	NULL

- *Friends* – Every user has the option to add other users to their friend list. This will enable them to play games together. Every user's friend list is supposed to be stored in this table.

  Columns: Game_ID (Primary Key), Name, Price, Release Date

  Table Creation:

```
CREATE TABLE `friend_list` (
  `Relationship ID` int NOT NULL AUTO_INCREMENT,
  `user_id` int NOT NULL,
  `friend_id` int NOT NULL,
  PRIMARY KEY (`Relationship ID`),
  KEY `user_ids_idx` (`user_id`),
  KEY `friend_ids_idx` (`friend_id`),
  CONSTRAINT `friend_ids` FOREIGN KEY (`friend_id`) REFERENCES `users` (`User_ID`),
  CONSTRAINT `user_ids` FOREIGN KEY (`user_id`) REFERENCES `users` (`User_ID`)
```

) COMMENT='Table which contains the list of all the friends a user has using the User_ID from the Users table'

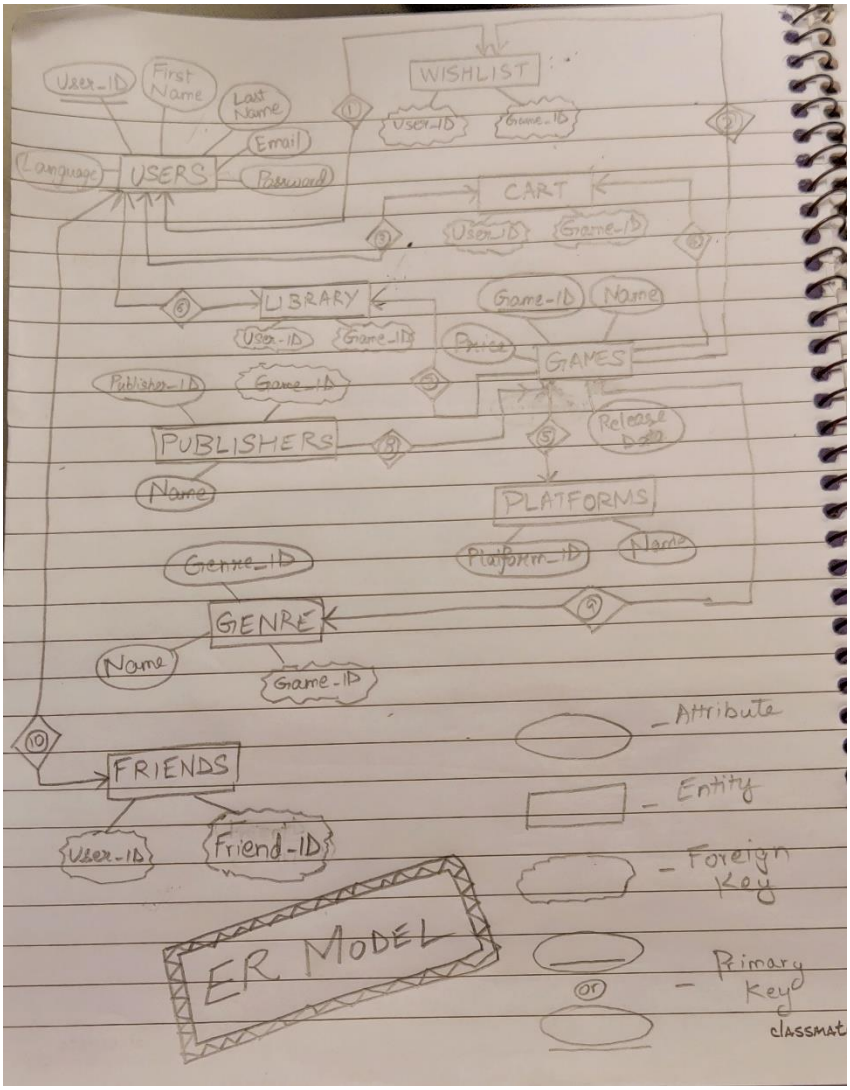
Table Population:

```
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('1', '1', '2')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('2', '1', '3')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('3', '1', '4')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('4', '1', '5')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('5', '2', '1')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('6', '2', '3')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('7', '2', '4')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('8', '3', '1')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('9', '3', '2')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('10', '4', '1')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('11', '4', '2')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('12', '5', '1')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('13', '5', '6')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('14', '6', '5')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('15', '4', '6')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('16', '6', '4')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('17', '6', '7')
INSERT INTO `server_database`.`friend_list` (`Relationship ID`, `user_id`, `friend_id`) VALUES ('18', '7', '6')
```

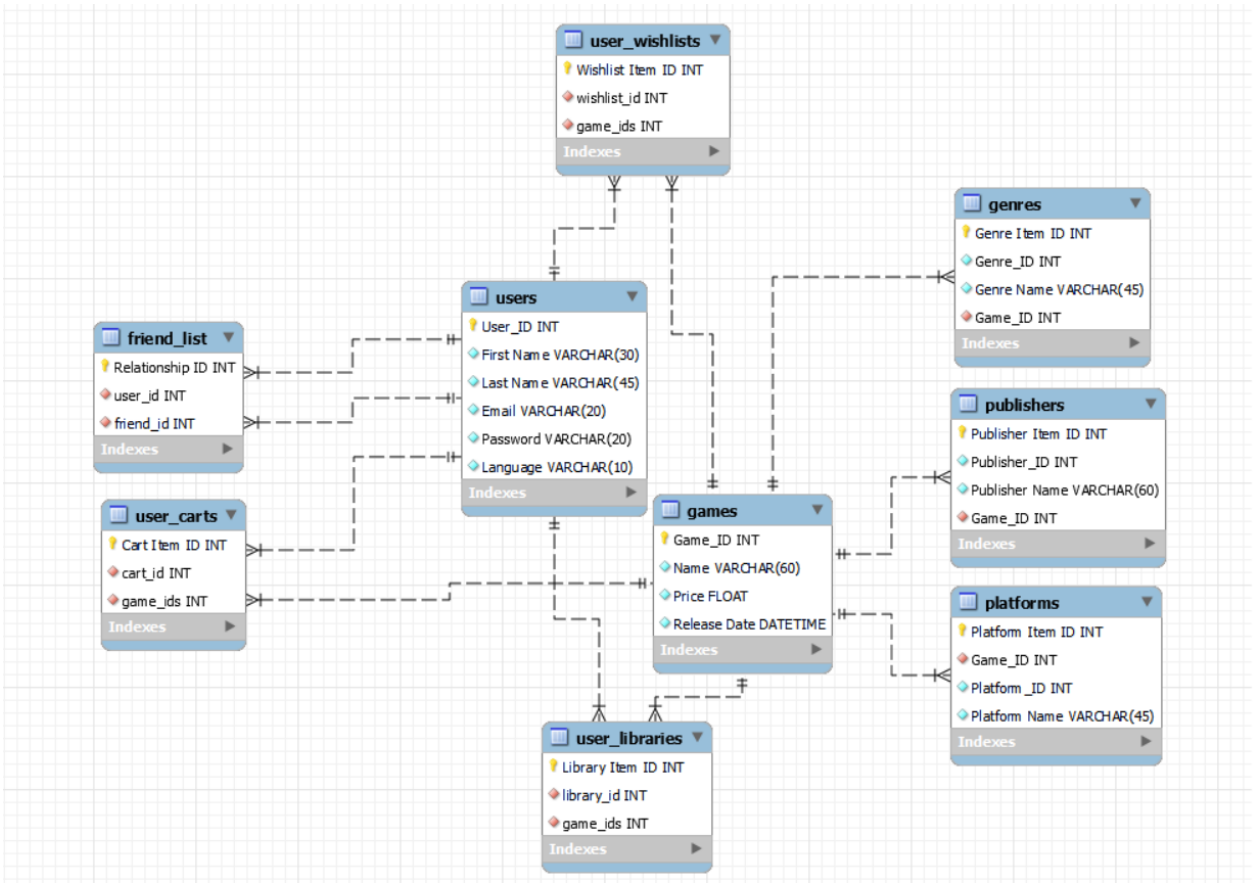
Final Table:

	Relationship ID	user_id	friend_id
▶	1	1	2
	2	1	3
	3	1	4
	4	1	5
	5	2	1
	6	2	3
	7	2	4
	8	3	1
	9	3	2
	10	4	1
	11	4	2
	12	5	1
	13	5	6
	14	6	5
	15	4	6
	16	6	4
	17	6	7
	18	7	6
*	NULL	NULL	NULL

ENTITY RELATION MODEL



FUNCTIONAL DEPENDENCIES (as generated by the SQL Workbench)



One to One relations

One to Many relations

Many to Many relations

TRIGGERS

- *Check whether the cost of the game entered is mistakenly entered negative*
  Solution: Maybe the (-ve) sign was mistakenly entered. So, the cost is converted to its positive counterpart using ABS () function. This is done BEFORE INSERT ON way.

  Code:

```
DROP TRIGGER IF EXISTS `games`.`check_price`;
        CREATE TRIGGER `check_price` BEFORE INSERT ON `games` FOR
        EACH ROW BEGIN
                IF NEW.price < 0
                        THEN SET NEW.price = abs(NEW.price);
        END IF;
END
```

VIEWS

- *Retrieving the User IDs of those users who have purchased GTA 5 (using Game ID = 3) and have it in their Library*

  Code:

```
CREATE VIEW `gta5_holders` AS
   SELECT
      `user_libraries`.`library_id` AS `GTA 5 Holders' User ID`
   FROM
      `user_libraries`
   WHERE
      (`user_libraries`.`game_ids` = 3)
```

  Output:

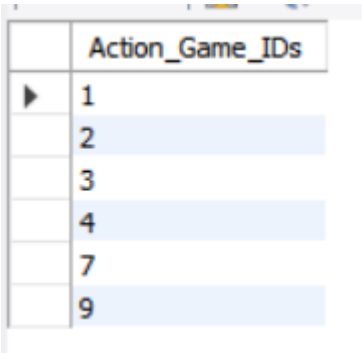
GTA 5 Holders' User ID
1
4
7

- *Retrieving the Game IDs of those games whose genre is 'Action'*

  Code:

```
CREATE VIEW `action_games` AS
   SELECT
      `genres`.`Game_ID` AS `Action_Game_IDs`
   FROM
      `genres`
   WHERE
      (`genres`.`Genre Name` = 'Action')
```

Output:



Action_Game_IDs
1
2
3
4
7
9

DATA AND SECURITY

Data Integri