Relational Databases CA
Design Document

Student Details

Name: Daniels Nagornuks

Number: 20106595

Group: W2 (Cloud Computing)

Year: 2

Table of Contents

Business Description

The business that will utilise relational database is called Dan's Games Store. This is a chain of stores that sells video game disks across the Republic of Ireland. The business will use SQL to store various information, specifically regarding the games, game disks, distributors, stores and employees. The business will not use SQL to store customer orders, or information about the customers themselves.

There will be several stores in Ireland, and each store would employ several employees. The employees would only be able to work at a single store. Some employees would be managed by other employees (i.e. managers), who in turn would be managed by another employee.

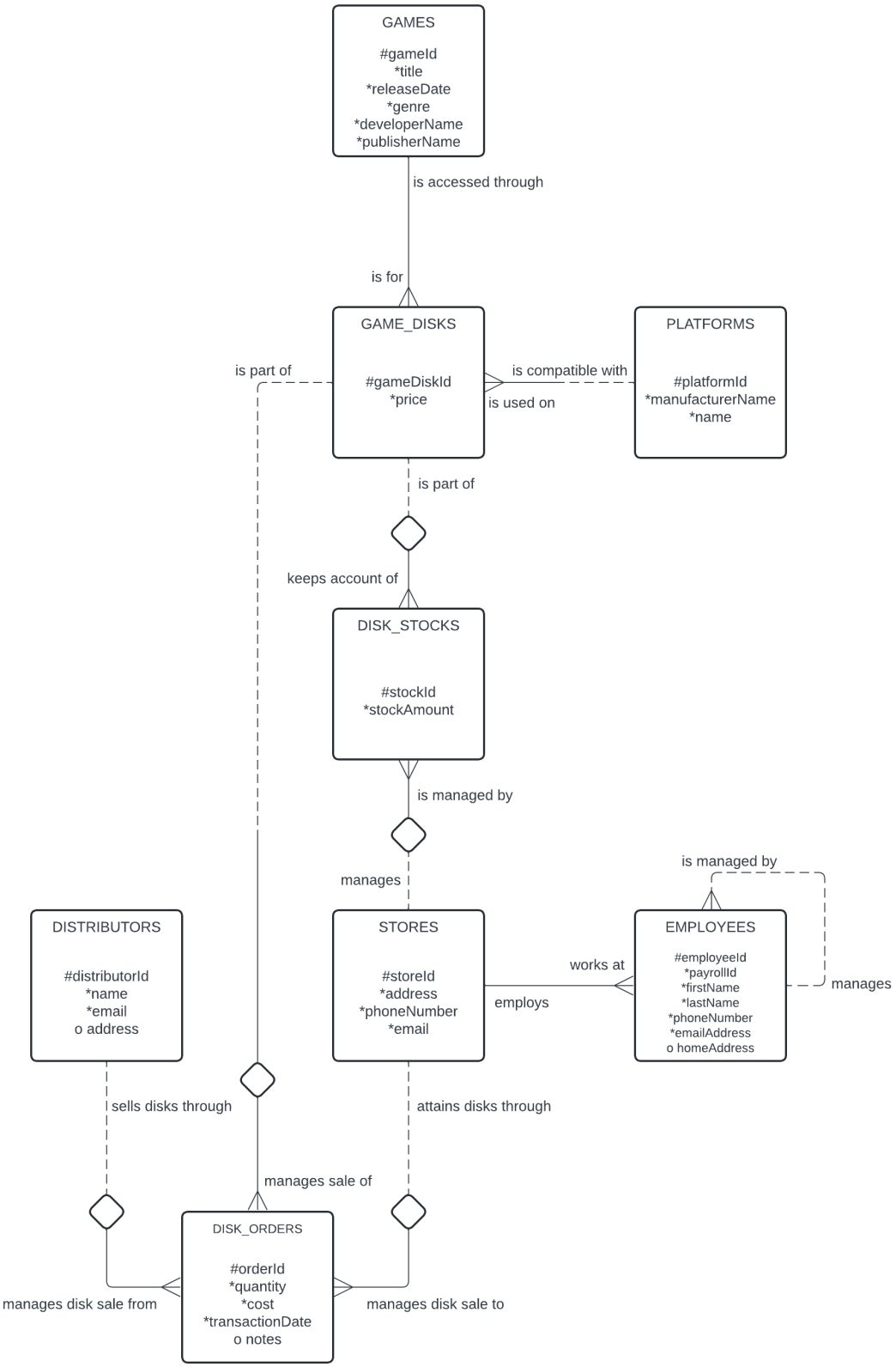
Information about games would be stored in the database, such as title, release date, genre, developer and publisher name. These games would be accessed in a disk form (from consumers' point of view). A game would have access to it through one or more disks (as there are several platforms that would support each disk, mostly PlayStation, Xbox, PC and Nintendo Switch), so a single disk would only be supported on a single platform, recorded as a separate entity in the database. Naturally, a game disk could only be for a single game.

Since disks for the same game and platform aren't unique from one another, it would make sense to have a disk stock entity that would be used to keep track of the quantity of a specific game disk at a specific store to lower processing overhead. A disk could of course have several stock lists, but a stock list would only keep track of a specific game disk. A store would have several stock lists, but a stock list, similarly as mentioned above, can only be managed by a single store.

A store would attain disks through disk orders. Store would have several disk orders made and recorded, and a disk order would only be for a specific store. The sellers from whom the disk orders are made, would be the distributors. A distributor could process many disk orders for stores, and a disk order would only be processed by a single distributor. Disk order would also relate to the game disk entity, to keep track of what game disks were purchased. A disk order would have an attribute called quantity (of a specific game disk) and would only take account of a single game disk. If in the real world, an order was made for several types of disks (such as maybe the same game but for different platforms), this order while being recorded as a one big order on the front end, would be split into several different smaller orders when recorded in the SQL database. Lastly, a game disk could be part of more than one disk order (in essence, a game disk isn't a specific tangible disk object, but more of a type of disk).

ChatGPT was used to clear up certain aspects, such as if certain entities/attributes were necessary to be stored. Not everything that was learned from the chat was applied to the design, while others were adapted. The link to the chat:
https://chatgpt.com/share/67190af0-cdb0-8002-8a4d-3db5420a24bc

EER Diagram


GAMES
#gameId
*title
*releaseDate
*genre
*developerName
*publisherName
is accessed through
is for
GAME_DISKS
#gameDiskId
*price
is part of
is compatible with
is used on
PLATFORMS
#platformId
*manufacturerName
*name
is part of
keeps account of
DISK_STOCKS
#stockId
*stockAmount
is managed by
manages
is managed by
DISTRIBUTORS
#distributorId
*name
*email
o address
STORES
#storeId
*address
*phoneNumber
*email
works at
employs
EMPLOYEES
#employeeId
*payrollId
*firstName
*lastName
*phoneNumber
*emailAddress
o homeAddress
manages
sells disks through
attains disks through
manages sale of
DISK_ORDERS
#orderId
*quantity
*cost
*transactionDate
o notes
manages disk sale from
manages disk sale to