|  |  |
| --- | --- |
|  | **DOKUZ EYLÜL UNIVERSITY**  **ENGINEERING FACULTY**  **DEPT. OF COMPUTER ENGINEERING** |

ONLINE GAME STORE

# CME 3201 Database Management Systems

# Term Project Report

Phase 3

2019-2020 FALL

2400000145 MEHMET HAMDİ USLU

2015510035 BERKE KADAM

## Introduction

**Online shopping** (sometimes known as e-tail from "electronic retail" or e-**shopping**) is a form of electronic commerce which allows consumers to directly buy goods or services from a seller over the **Internet** using a web browser. In our project we aim to serve electronic copies of computer games by a digital distribution service.

**Digital distribution** (also referred to as **content delivery**, **online distribution**, or **electronic software distribution** among others) is the delivery or distribution of digital media content such as audio, video, software and video games. The term online distribution is typically applied to freestanding products; downloadable add-ons for other products are more commonly known as **downloadable content**. With the advancement of network bandwidth capabilities, online distribution became prominent in the **21st century**, with prominent platforms such as Steam and Epic Store and much more.

Wide spread of internet and increasing internet speed around the world accessibility of online game stores has risen as stated and with this accessibility. Rise of **digital copy sales** increased massively. This leap has become reason for reducing of sales as hard copies from stores. Reduce of hard copy sales had positive impact for companies because for the hard copy sales, companies had to spent massive amount of source on hard copy sale. Logistics, raw material used for hard copy cases, instructions, paper for case visibility; all this was an expense from their profit which they did not get until some point after release. Thanks to digital distribution all this effort became **couple megabyte** in digital world and saved companies millions, reduced earth materials used and reduced carbon footprint of the industry and most importantly let people free of storage place for this massive amount of hard copies.

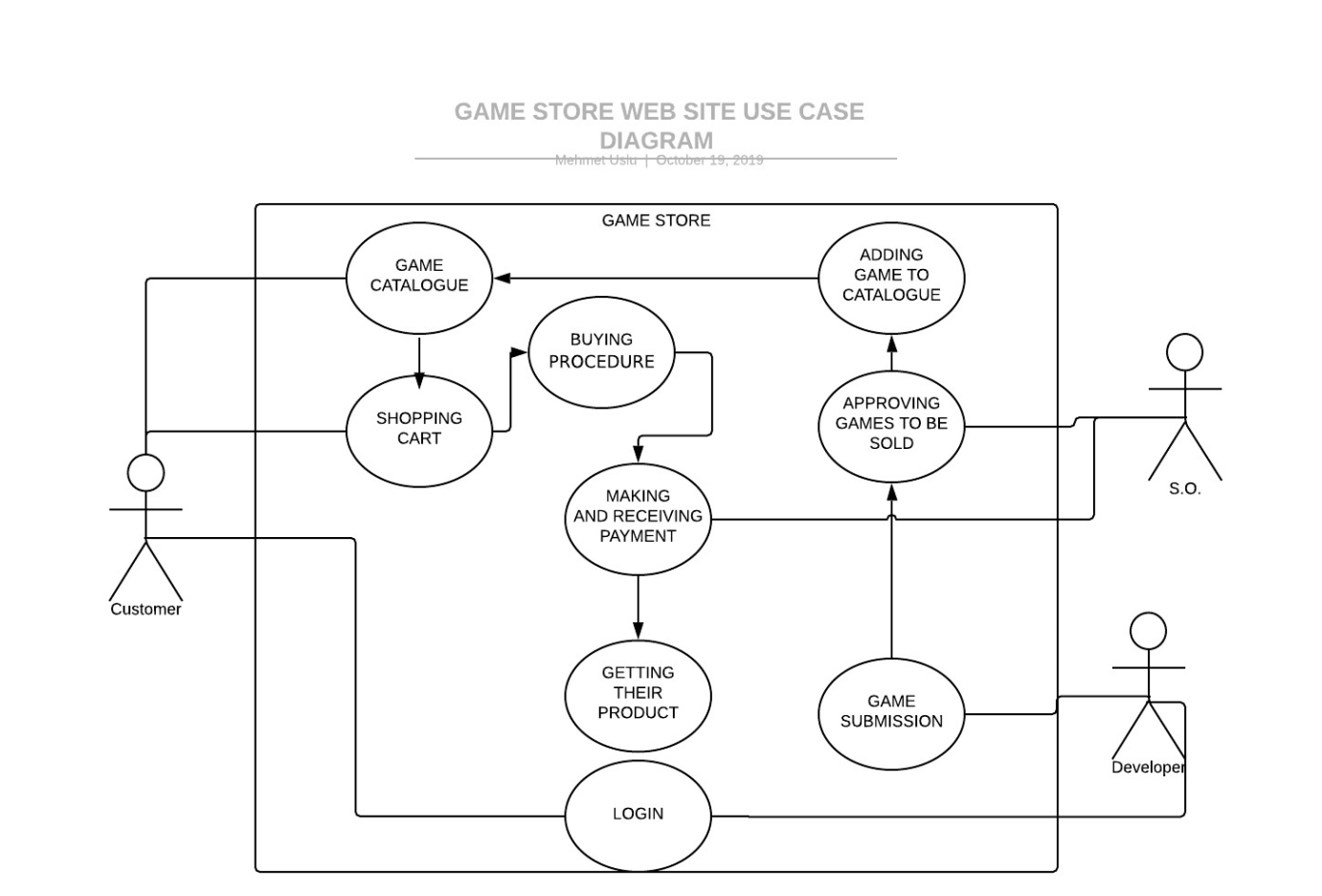
For this reason, in our project, we chose to develop an **online game purchase site**. These systems, which eliminate traditional hardcopy game sale system, combine game developers and players under **one platform**, making games easier to distribute. At the same time, it offers a market for **independent developers**, where they can easily show themselves.

Internet, video games and online shopping are more common among the younger generation, as we all know. As we mentioned in the report, the widespread use of Internet and digital distribution effect the Z generation most. Therefore, the interface of the project needs to be designed in such a way that young people can easily use it. In addition, the fact that young people have access to the game so easily creates a responsibility for our system. For this reason, we attach great importance to parental control of games the age limit and the content it contains will be included in the description of each game for parents to review.

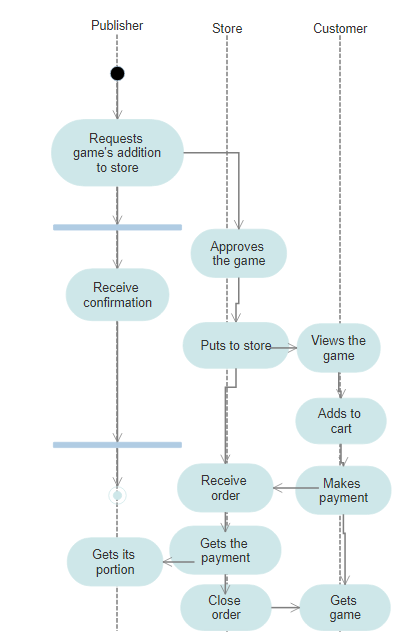
As we mentioned in the introduction, online gaming sales is a market that should not be overlooked, and we have prepared this report for our project named **GamerMarket**. This report states the mechanism of the online purchasing system and the users of the system, can be change in the future because of changes in requirements, is a critical roadmap for the progress of the project.

## Overview

System will enable users to examine and buy games and let publisher showcase and sell their games to our platform that we gain percentage of profit from their retail. Our main purpose in this project to create an online game store that user can examine and buy games and create a marketplace for publishers to serve their product and compete with another publishers.  
 There are three types of stakeholders; Customer, Publishers and Store Owner.  
**Customer**: Role of customer is to obtain their needs which is in this case to buy games. Customer can view the details of a game such as its genre, release date, in-game photos, parental advisory and review scores.  
**Publisher**: Role of publisher is to answer the need of their fans/customers; that need is to serve their product to target audience and profit from their long term project.  
**Store Owner**: Role of Store Owner (S.O.) is to fulfill needs of both customers and publishers. S.O. must enable publisher and customer to meet at a common ground. Let publishers compete & sell their games and let customers to buy their desired product.



**Activity Diagram**



## Assumptions/Constraints/Risks

### Assumptions

We’re going to develop a website by using .NET Web Application for front-end and SQL for database system. We will decide whether to use MSSQL or PostgreSQL after some reviews. Our website doesn’t effect of OS or end-user characteristics.

### Constraints

We don’t have much constraints. Our website can’t act outside of its gaming domain. There are only video games and users to view and buy them. So our constraints are: A user can only buy one copy of the game and cannot buy games that do not meet the age limit.

### Risks

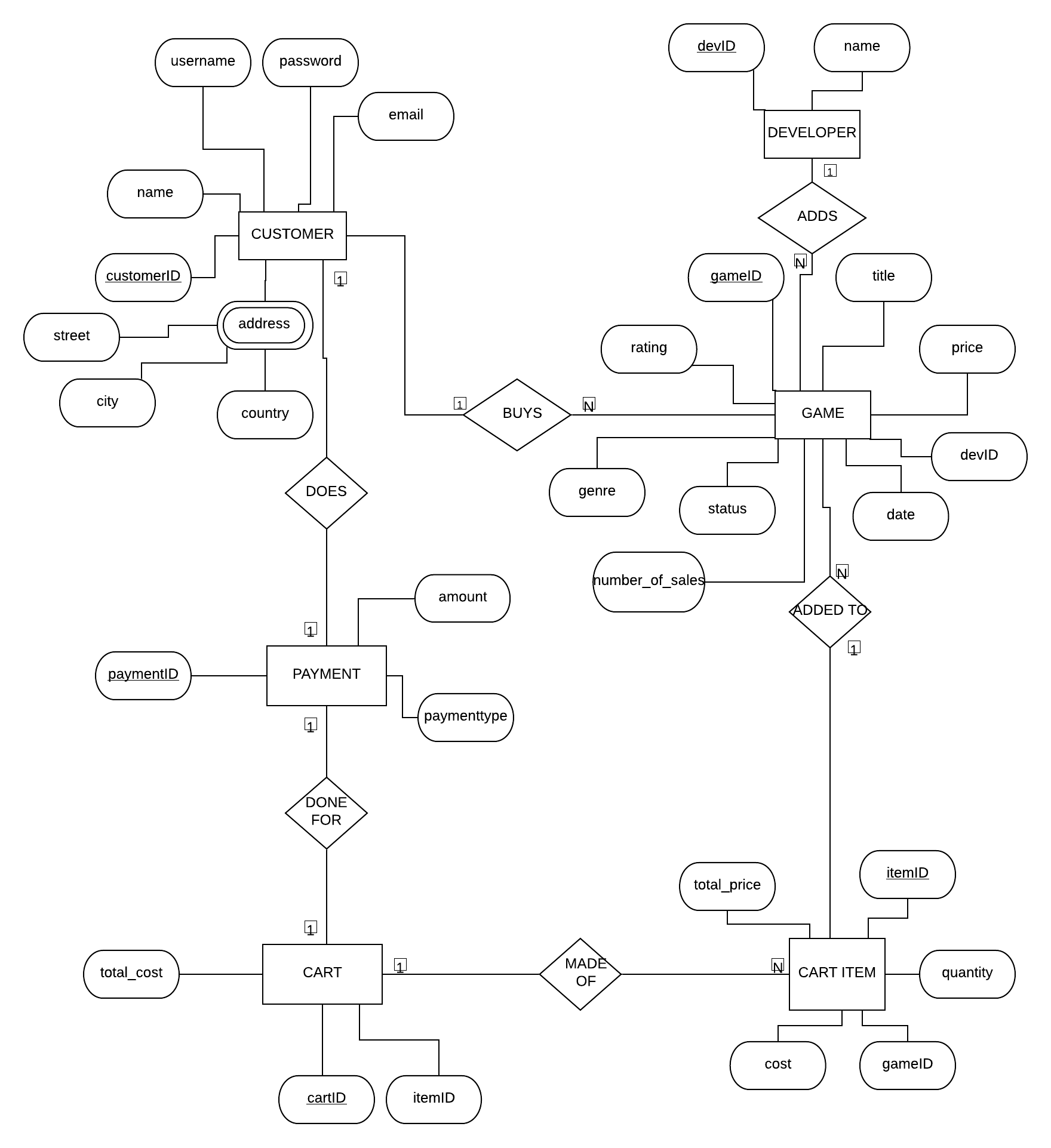
Users are responsible for the compatibility of the game they purchase with their own computer software and hardware. Although the details are mentioned in the game description, unconscious users pose a risk to us.

## Software Architecture

Instructions: Show your details of software architecture. Do you use n-tier architecture? How did you design it in your project? You may draw software architecture diagram (It depicts how a typical software system might interact with its users, external systems, data sources, and services.). How did you provide a connection between database and frontend? Give details about your software architecture.

## Detailed System Design

### Entity-Relationship Diagram

**

### Relational Algebra Expressions

SELECT G.Title, G.Price AS PriceList,

FROM Game G,

WHERE G.Price = 100

**π title, price (σ price=1005 (Game))**

SELECT G.Title,

FROM Developers D

INNER JOIN Games G ON D.DevID = G.DevID

WHERE D.DevName = ‘EA GAMES’

ORDER BY G.Title

**π title (σ DevID=’EA Games’ (Developer) ⨝Developer.DevID = Game.GameID Games)**

SELECT G.Title,

FROM Game G,

GROUP BY G.GameID

HAVING G.Rating > 70

**π title (σ rating > 70 (Game))**

SELECT G.Title,

FROM Game G,

WHERE G.Price > 100 AND Release Date < 2019

ORDER BY G.ReleaseDate ASC

**π title (σ price > 100 ^ release date < 2019 (Game))**

SELECT D.Name, AVG(G.Rating),

FROM Game G,

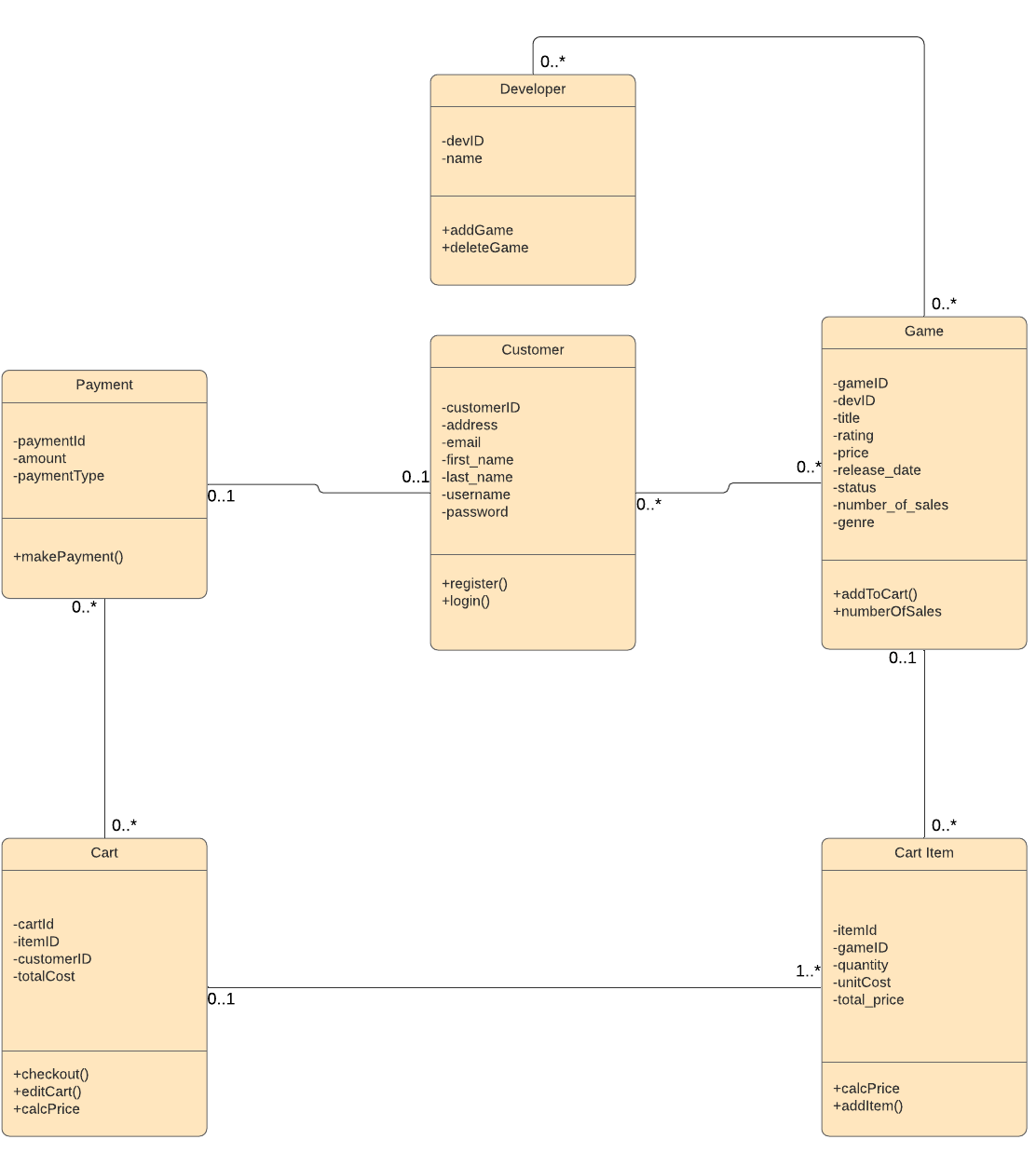
INNER JOIN Developers D ON D.DevID = G.DevID,

GROUP BY D.Name,

ORDER BY D.Name

**π developer.name** (**σ avg(rating)((σ devName (Developer) ⨝ Developer.DevID = Game.DevID Game)))**

### Class Diagram



### CRUD Matrix

For User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Relations\Operations** | **Create** | **Read** | **Update** | **Delete** |
| Game |  | X |  |  |
| Cart | X | X | X | X |
| CartItem |  | X |  |  |

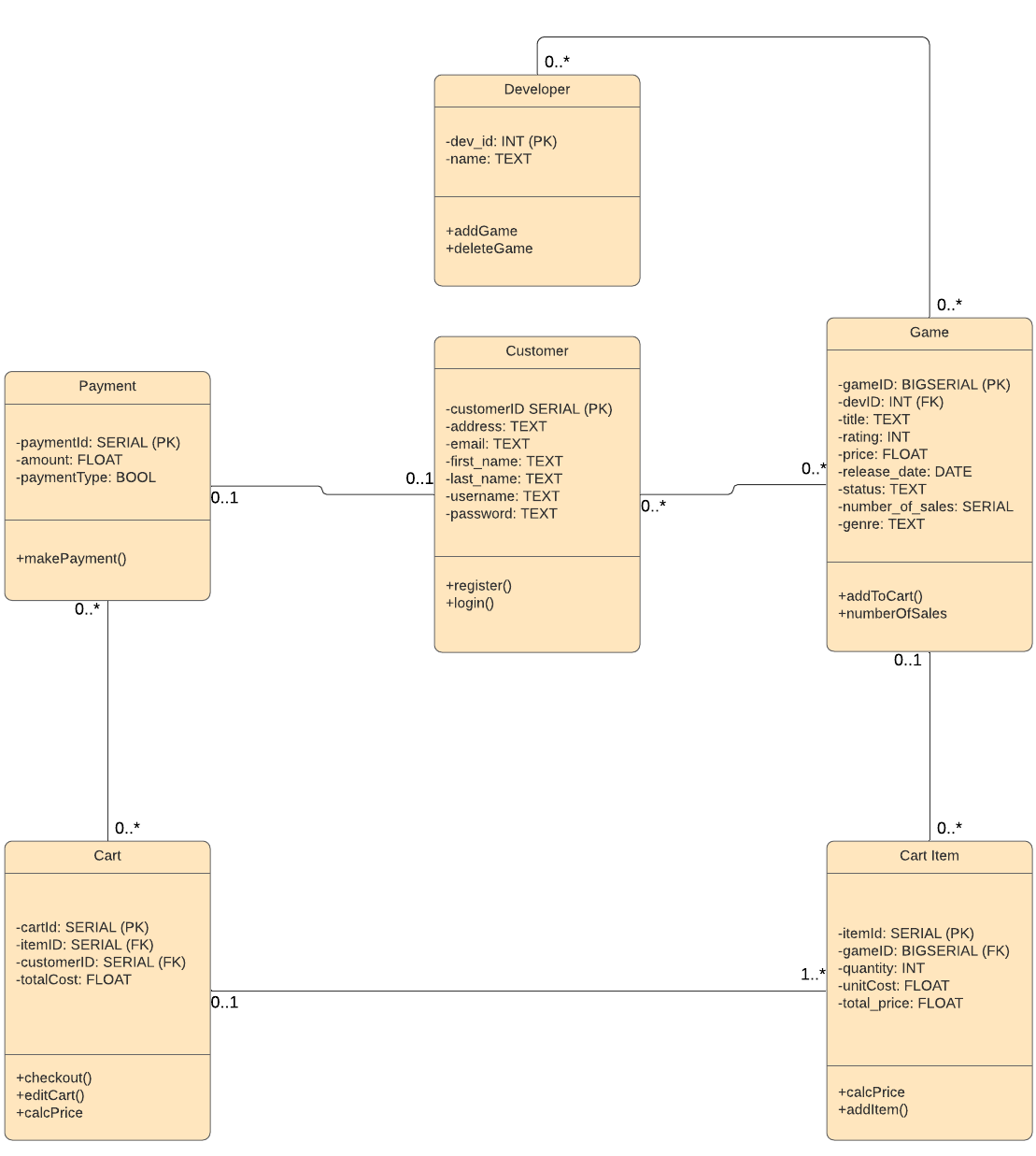
For Store Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Relations\Operations** | **Create** | **Read** | **Update** | **Delete** |
| Game | X | X | X | X |
| Cart |  | X |  |  |
| Customer |  | X | X | X |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Relations\Operations** | **Create** | **Read** | **Update** | **Delete** |
| Game | X | X | X | X |

For Developer

### Database Schema



**Appendix A: SQL Statements**

SET search\_path to game\_store\_app;

SET ROLE game\_store\_app;

CREATE TABLE customer (

customer\_id SERIAL PRIMARY KEY,

first\_name TEXT NOT NULL,

last\_name TEXT NOT NULL,

email TEXT NOT NULL UNIQUE,

username TEXT NOT NULL,

address TEXT NOT NULL,

password TEXT NOT NULL,

CHECK(first\_name !~ '\s' AND last\_name !~ '\s'),

CHECK (email ~\* '^\w+@\w+[.]\w+$'),

CHECK (char\_length(password)>=8)

);

CREATE TABLE developer (

dev\_id INT PRIMARY KEY,

name TEXT NOT NULL

);

CREATE TABLE game (

game\_id BIGSERIAL PRIMARY KEY,

dev\_id INT NOT NULL REFERENCES developer(dev\_id),

title TEXT NOT NULL,

release\_date DATE NOT NULL,

genre TEXT NOT NULL,

rating INT NOT NULL,

number\_of\_sales SERIAL NOT NULL,

price FLOAT NOT NULL

);

CREATE TABLE cart\_item

(

item\_id SERIAL PRIMARY KEY,

game\_id BIGSERIAL NOT NULL REFERENCES game(game\_id),

quantity INT NOT NULL,

unit\_cost FLOAT NOT NULL,

total\_price FLOAT NOT NULL

);

CREATE TABLE cart (

cart\_id SERIAL PRIMARY KEY,

item\_id SERIAL NOT NULL REFERENCES cart\_item(item\_id),

customer\_id SERIAL NOT NULL REFERENCES customer(customer\_id),

total\_cost FLOAT NOT NULL

);

CREATE TABLE payment(

payment\_id SERIAL PRIMARY KEY,

amount FLOAT NOT NULL,

payment\_type BOOLEAN NOT NULL

);

CREATE TABLE admin (

admin\_id SERIAL PRIMARY KEY,

email TEXT NOT NULL UNIQUE,

username TEXT NOT NULL,

password TEXT NOT NULL,

CHECK (email ~\* '^\w+@\w+[.]\w+$'),

CHECK (char\_length(password)>=8)

);