УО «Белорусский государственный университет информатики и радиоэлектроники»

Кафедра ПОИТ

Отчёт по лабораторной работе №6

По предмету

Базы данных

Выполнил:

Гладкий М.Г.

Проверил:

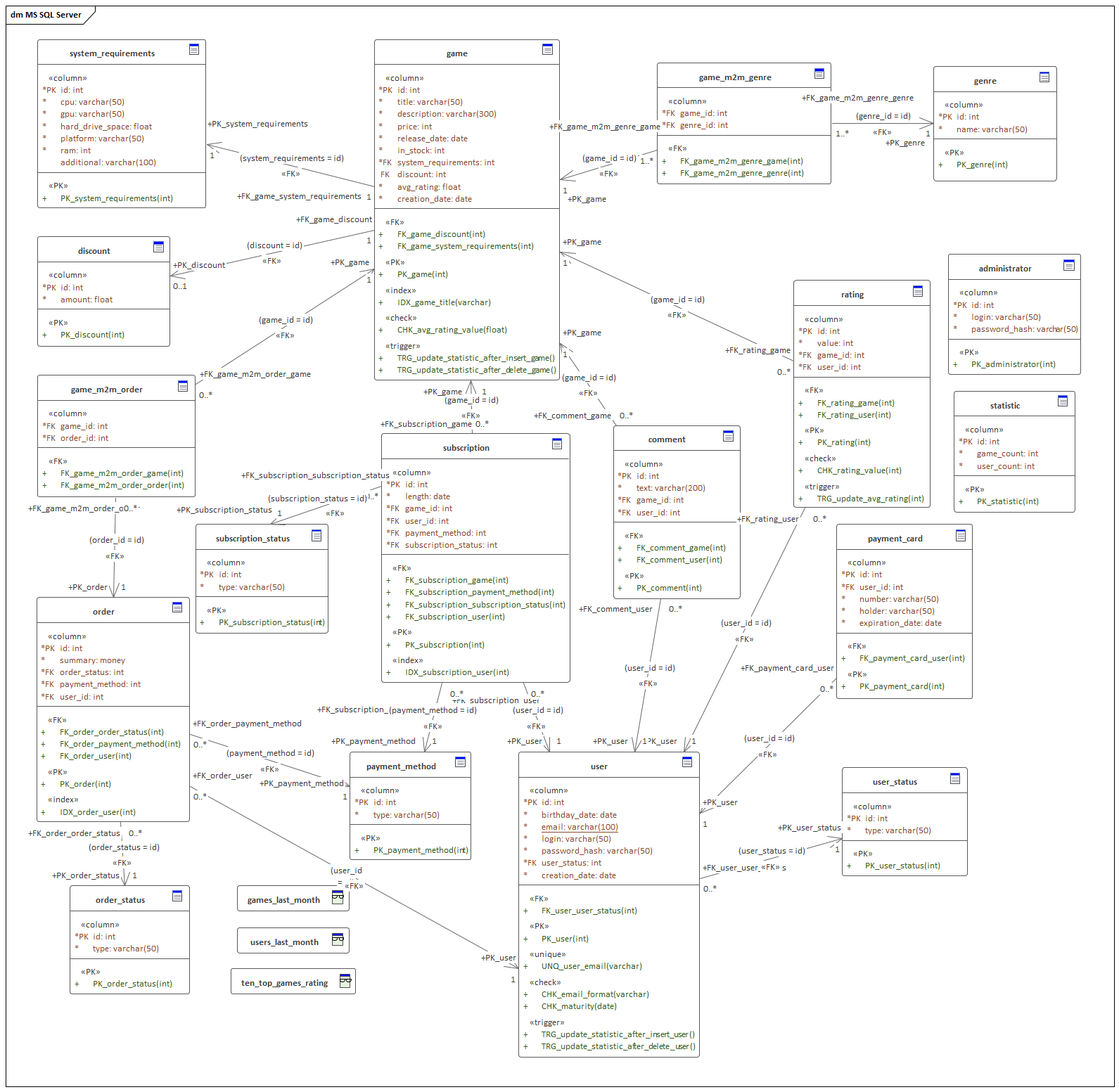
Фадеева Е.Е.

Группа 851005

Минск 2021

Предметная область: Интернет-магазин компьютерных игр

Даталогическая модель:



1) Каскадные операции:

Были настроены следующие каскадные операции:

FK\_game\_discount: ON DELETE – No action, ON UPDATE – Cascade;

FK\_game\_system\_requirements: ON DELETE – No action, ON UPDATE – Cascade;

FK\_game\_m2m\_genre\_game: ON DELETE – Cascade, ON UPDATE – Cascade;

FK\_ game\_m2m\_genre\_genre: ON DELETE – Cascade, ON UPDATE – Cascade;

FK\_game\_m2m\_order\_game: ON DELETE – Cascade, ON UPDATE – Cascade;

FK\_ game\_m2m\_order\_order: ON DELETE – Cascade, ON UPDATE – Cascade;

FK\_order\_order\_status: ON DELETE – No action, ON UPDATE – Cascade;

FK\_order\_payment\_method: ON DELETE – No action, ON UPDATE – Cascade;

FK\_order\_user: ON DELETE – No action, ON UPDATE – Cascade; Заказы должны оставаться в базе данных для правильного подсчета средств.

FK\_subscription\_game: ON DELETE – No action, ON UPDATE – Cascade; Подписки остаются в БД всегда.

FK\_subscription\_payment\_method: ON DELETE – No action, ON UPDATE – Cascade;

FK\_subscription\_subscription\_status: ON DELETE – No action, ON UPDATE – Cascade;

FK\_subscription\_user: ON DELETE – No action, ON UPDATE – Cascade;

FK\_comment\_game: ON DELETE – Cascade, ON UPDATE – Cascade;

FK\_comment\_user: ON DELETE – Cascade, ON UPDATE – Cascade;

FK\_rating\_game: ON DELETE – Cascade, ON UPDATE – Cascade;

FK\_rating\_user: ON DELETE – Cascade, ON UPDATE – Cascade;

FK\_user\_user\_status: ON DELETE – No action, ON UPDATE – Cascade;

FK\_payment\_card\_user: ON DELETE – Cascade, ON UPDATE – Cascade;

2) Представления:

Выборка 10 игр с высочайшим рейтингом

CREATE VIEW 'ten\_top\_games\_rating' AS

SELECT **\*** FROM 'game'

ORDER BY 'avg\_rating' DESC

LIMIT 10

Выборка игр, добавленных за последний месяц

CREATE VIEW 'games\_last\_month' AS

SELECT **\*** FROM 'game' WHERE DATE**(**'creation\_date'**)** **>** **(**CURDATE**()** **-** INTERVAL 1 MONTH**)**

Выборка пользователей, зарегистрированных за последний месяц

CREATE VIEW 'users\_last\_month' AS

SELECT **\*** FROM 'user' WHERE DATE**(**'creation\_date'**)** **>** **(**CURDATE**()** **-** INTERVAL 1 MONTH**)**

3) Проверки:

Валидность email пользователя

ALTER TABLE 'user'

ADD CONSTRAINT 'CHK\_email\_format'

CHECK **(**'email' REGEXP '^[^@][+@[^@]+\.[^@]+$](mailto:+@[%5e@]+\.%5b%5e@%5d+$)'**)**

Проверка рейтинга на значения от 1 до 10

ALTER TABLE 'rating'

ADD CONSTRAINT 'CHK\_rating\_value'

CHECK **(**'value' BETWEEN 1 AND 10**)**

ALTER TABLE 'game'

ADD CONSTRAINT 'CHK\_avg\_rating\_value'

CHECK **(**'avg\_rating' BETWEEN 1 AND 10**)**

Пользователь совершеннолетний

ALTER TABLE 'user'

ADD CONSTRAINT 'CHK\_maturity'

CHECK DATEDIFF**(**CURDATE**(),** DATE**(**'birthday\_date'**))** **/** 365 **>=** 18

4) Триггеры:

Обновление среднего рейтинга игры после вставки, обновления или удаления рейтинга, поставленного пользователем

CREATE TRIGGER `TRG\_update\_avg\_rating` AFTER INSERT**,** UPDATE**,** DELETE ON `**rating**`

FOR EACH ROW BEGIN

UPDATE `**game**` SET

`avg\_rating` **=** SELECT AVG**(**`value`**)** FROM `**rating**` WHERE `game\_id` **=** OLD**.**`game\_id`**,**

WHERE `id` **==** OLD**.**`game\_id`**;**

END**;**

Обновление статистики после вставки/удаления пользователя

CREATE TRIGGER `TRG\_update\_statistic\_after\_insert\_user` AFTER INSERT ON `user`

FOR EACH ROW BEGIN

UPDATE `**statistic**` SET `user\_count` **=** `user\_count` **+** 1

END**;**

CREATE TRIGGER `TRG\_update\_statistic\_after\_delete\_user` AFTER DELETE ON `user`

FOR EACH ROW BEGIN

UPDATE `**statistic**` SET `user\_count` **=** `user\_count` **-** 1

END**;**

Обновление статистики после вставки/удаления игры

CREATE TRIGGER `TRG\_update\_statistic\_after\_insert\_game` AFTER INSERT ON `**game**`

FOR EACH ROW BEGIN

UPDATE `**statistic**` SET `game\_count` **=** `game\_count` **+** 1

END**;**

CREATE TRIGGER `TRG\_update\_statistic\_after\_delete\_game` AFTER DELETE ON `**game**`

FOR EACH ROW BEGIN

UPDATE `**statistic**` SET `game\_count` **=** `game\_count` **-** 1

END**;**