Заказчик: Стриминговый сервис для просмотра сериалов и фильмов.

Задача: Реализовать базу данных, хранящую информацию о пользователях сервиса и фильмах, представленных на нем.

- 1) Данные об аккаунтах пользователей (id пользователя на сервисе, email, ФИО, пароль, номер телефона, идентификатор подписки)
- 2) Действия пользователей (фильмы и сериалы, которые они посмотрели/не посмотрели/планируют посмотреть, комментарии к фильмам/сериалам)
- 3) Информация о фильмах/сериалах, представленных на сервисе (год производства, страна, жанр, режиссер, список актеров)

Реализация: используется формат реляционной БД. Реализуются следующие 13 таблиц (11 для сущностей и 2 для связей):

- 1) <u>User</u> информация о пользователях
- 2) Subscription информация о подписке
- 3) Comments комментарии, оставляемые пользователем
- 4) <u>Ratings</u> оценки, поставленные фильмам
- 5) <u>Usage history</u> какой фильм пользователь посмотрел/не посмотрел
- 6) Status of video фильм просмотрен/не просмотрен
- 7) Video информация об имеющимся контенте
- 8) <u>Video types</u> жанр и тип (сериал, фильм или другое)
- 9) Countries страна-производитель
- 10) Actors информация об актерах
- 11) <u>Directors</u> информация о режиссерах
- 12) <u>Shoot</u> идентификатор режиссера для каждого фильма
- 13) <u>Filming</u> идентификатор актера для каждого фильма

Такая декомпозиция базы данных обеспечивает, что каждая таблица находится в 3NF.

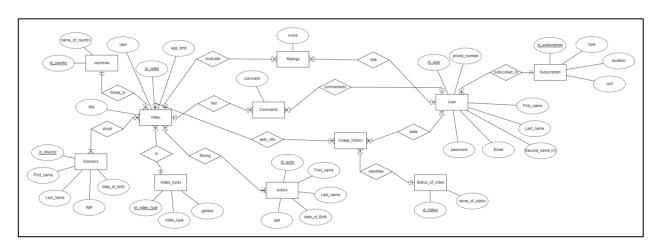
Описание сущностей:

Название	Название	Тип данных	Описание	Первичный(РК)/
сущности	атрибута			Внешний ключи
				(FK)
Actors	Id_actor	int	Идентификатор	PK
			актера	
	First_name	varchar(60)	Имя	
	Last_Name	varchar(60)	Фамилия	
	Date_of_birth	date	Дата рождения	
	age	int	возраст	
Comments	comments	varchar (400)	Текст комментария	
			к фильму	
	Id_user	int	Идентификатор	FK
			пользователя	
	Id_video	int	Идентификатор	FK
			видео	
Directors	Id_director	int	Идентификатор	PK
			режиссера	
	First_name	varchar(60)	имя	
	Last_name	varchar(60)	фамилия	
	Date_of_birth	date	Дата рождения	
	age	int	возраст	

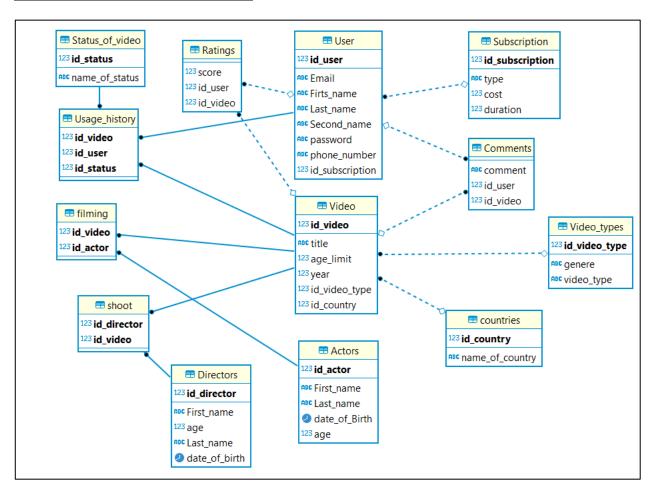
	score	Int	Выставленная	
			оценка	
	Id user	varchar(120)	Идентификатор	FK
Ratings	_		пользователя	
	Id_video	varchar(120)	Идентификатор	FK
	_		фидео	
Status_of_video	Name of stattus	int	Статус фильма по	
			просмотрам	
	Id status	varchar(50)	Идентификатор	PK
		, , ,	статуса	
	Id_subscription	int	идентификатор	PK
			подписки	
	type	Varchar(60)	Тип подписки	
Subscription	cost	int	стоимость	
			подписки	
	duration	int	Дата истечения	
			подписки	
	Id_of_video	int	Идентификатор	PK(FK)
			видео	
Usage history	Id_of_users	int	Идентификатор	PK(FK)
Osage_mstory			пользователя	
	Id_of_status	int	Идентификатор	PK(FK)
			статуса контента	
	Id_user	int	идентификатор	PK
			пользователя	
	email	Varchar(60)	Электронная почта	
	First_name	Varchar(60)	Р В В В В В В В В В В	
	Last_name	Varchar(60)	фамилия	
User	Second_name	Varchar(60)	отчество	
	password	Varchar(60)	Пароль	
			пользователя	
	Phone_number	Varchar(60)	Номер телефона	
	Id_subscription	int	Идентификатор	FK
			подписки	
	Id_video	int	Идентификатор	PK
		77 1 (60)	видео	
	title	Varchar(60)	Название фильма	
	Age_limit	int	Возрастное	
Video		1 .	ограничение	
	year	date	Год выпуска	FIZ
	Id_video_type	int	Идентификатор	FK
	Id accomptant	int	типа видео	FK
	Id_country	int	Идентификатор	r K
Video_types	Id video type	int	страны Идентификатор	PK
	Id_video_type	1111	типа видео	1 13
	genere	Varchar(60)	Жанр видео	
	Video type	Varchar(60)	Название типа	
	v raco_type	v archar(00)	видео	
Countries	Id country	int	Идентификатор	PK
	Id_country	1111	страны	118
	Name of country	Varchar(60)	Название страны	
		, archar(00)	тазвание страны	<u> </u>

	Id_video	int	Идентификатор	FK
Filming			видео	
	Id_actor	Varchat(60)	Название страны	FK
Shoot	Id_director	int	Идентификатор	FK
			режиссера	
	Id_video	int	Идентификатор	FK
			видео	

ER-диаграмма базы данных:



class diagram UML базы данных:



Ограничения: Подразумевается, что для корректной работы базы данных значения каждого поля каждой сущности обязаны быть ненулевыми.

- 1)Уникальные идентификаторы (id) по умолчанию обладают этим свойством.
- 2) Для всех неидентификационных полей, относящихся к сущностям фильмов, режиссеров и актеров (например, имя режиссера, год выпуска фильма, возраст актера и т.д.) выполнение условия очевидно (данная информация находится в открытом доступе).
- 3) Для пользовательской информации (email, пароль и т.д.) «notnull-ность» должна обеспечиваться на этапе регистрации аккаунта (то есть в случае БД на этапе insert'a).
- 4) Для сущностей Ratings и Comments: если запись о рейтинге/комментарии существует, то очевидно, что оценка/комментарий не пустое поле.
- 5) Для таблиц, описывающих связи (shoot, filming) выполнение свойства тривиально (их поля id'шники => ненулевые из п.1)