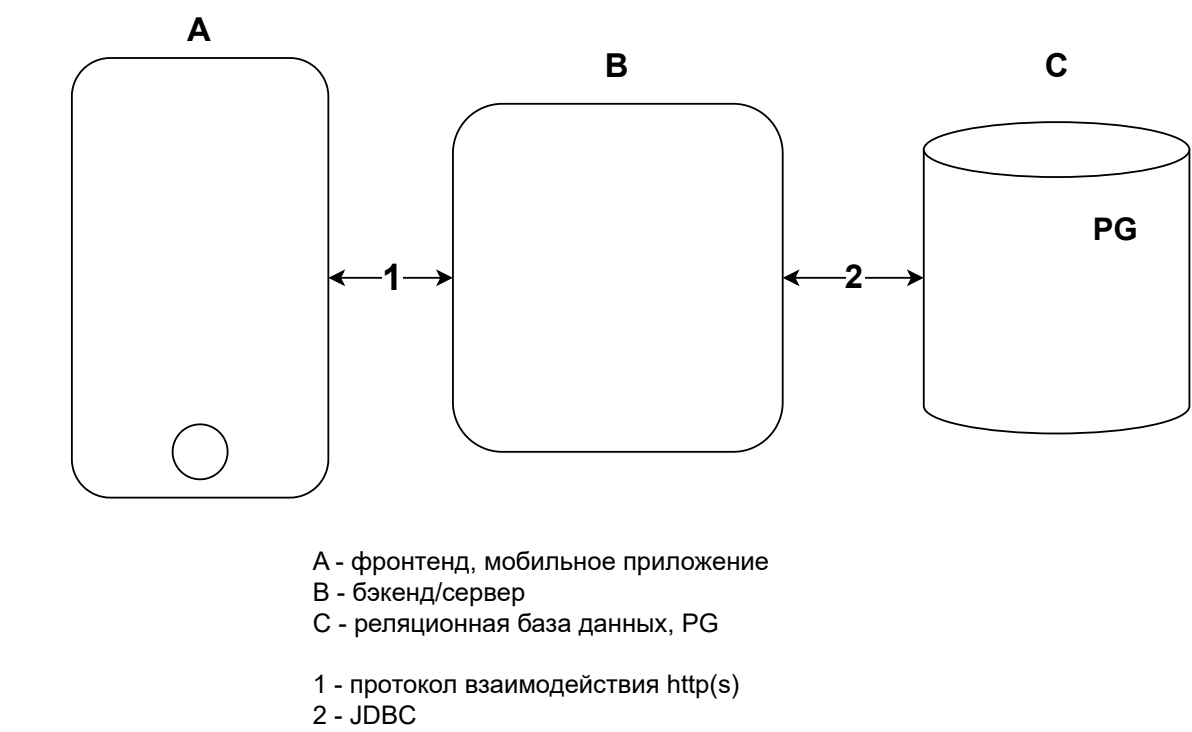


# 1. Архитектура

Фронтенд – это схематичный веб или мобильное приложение системы. Также фронтенд называют клиентом

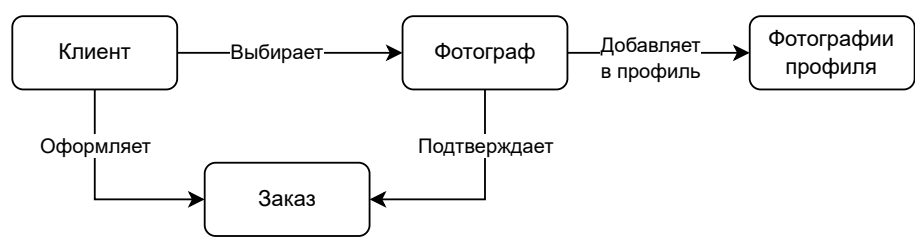
Бэкенд – это внутренний сервер системы проекта. Сервер обращается к базе данных с полученными данными от клиента

База данных – это хранилище данных, с которыми будет работать сервер.



# 2. Модель данных

Модель данных – это представление данных, атрибутного состава сущностей, как сущности связаны друг с другом. Объект Заказ выходит за рамки рассматриваемой user story, однако считаю полезным ее отобразить для понимания связи Клиент-фотограф в финальном сценарии.



Объект Клиент

Родительская сущность	Атрибут	Описание
Client	-----	Профиль клиента в приложении
	tel_number	Номер телефона
	name	Имя
	surname	Фамилия
	patronymic	Отчество
	city	Город проживания
	email	Почта для связи

Объект Photographer

Родительская сущность	Атрибут	Описание
<i>Photographer</i>	-----	Профиль фотографа
	tel_number	Номер телефона
	name	Имя
	surname	Фамилия
	patronymic	Отчество
	email	Почта для связи
	type	Тип съемок (портрет, лав-стори, семейная и т.п)
	style	Стиль фотографий (фэшн, минимализм, чб и тп)
	rating	Оценка клиентов (рейтинг)

Объект Заказ

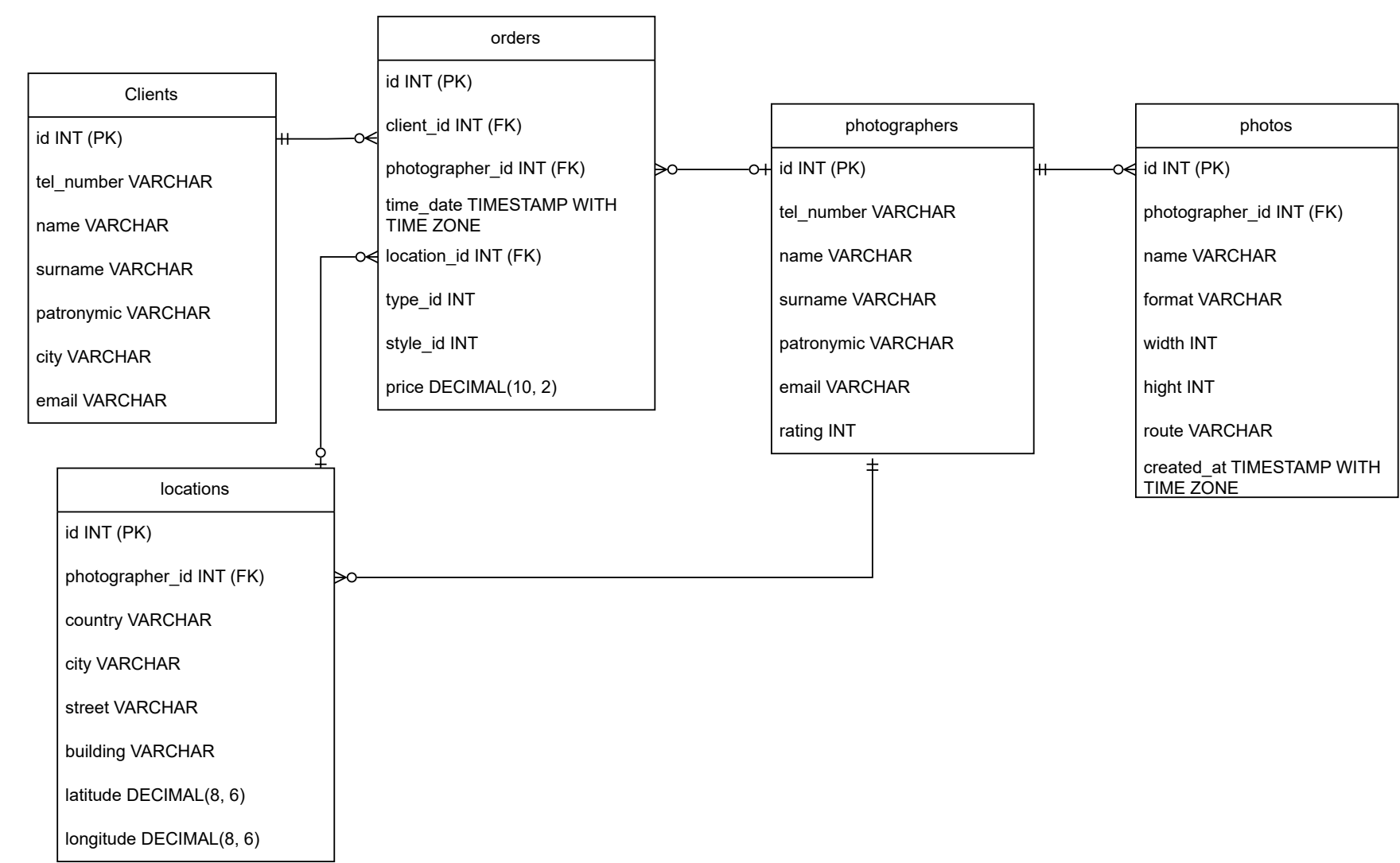
Дочерняя сущность	Атрибут	Описание
<i>Order</i>	-----	Заказ на фотосессию
	client	Кто заказчик фотосессии
	photographer	Кто фотограф фотосессси
	time_date	Дата и время проведения съемки
	location	Локация съемки
	type	Тип съемки (портрет, лав-стори и тп)
	style	Стиль съемки (фэшн, чб и тп)
	price	Итоговая стоимость заказа

Объект Фотографии

Дочерняя сущность	Атрибут	Описание
<i>Photos</i>	-----	Заказ на фотосессию
	photographer	Фотограф, которому принадлежит фото
	name	наименование файла фотографии
	format	формат фотографии (расширение)
	wight	ширина фото в пкс
	hight	высота фото в пкс
	route	расположение (адрес) фото
	created_at	дата добавления фото в профиль

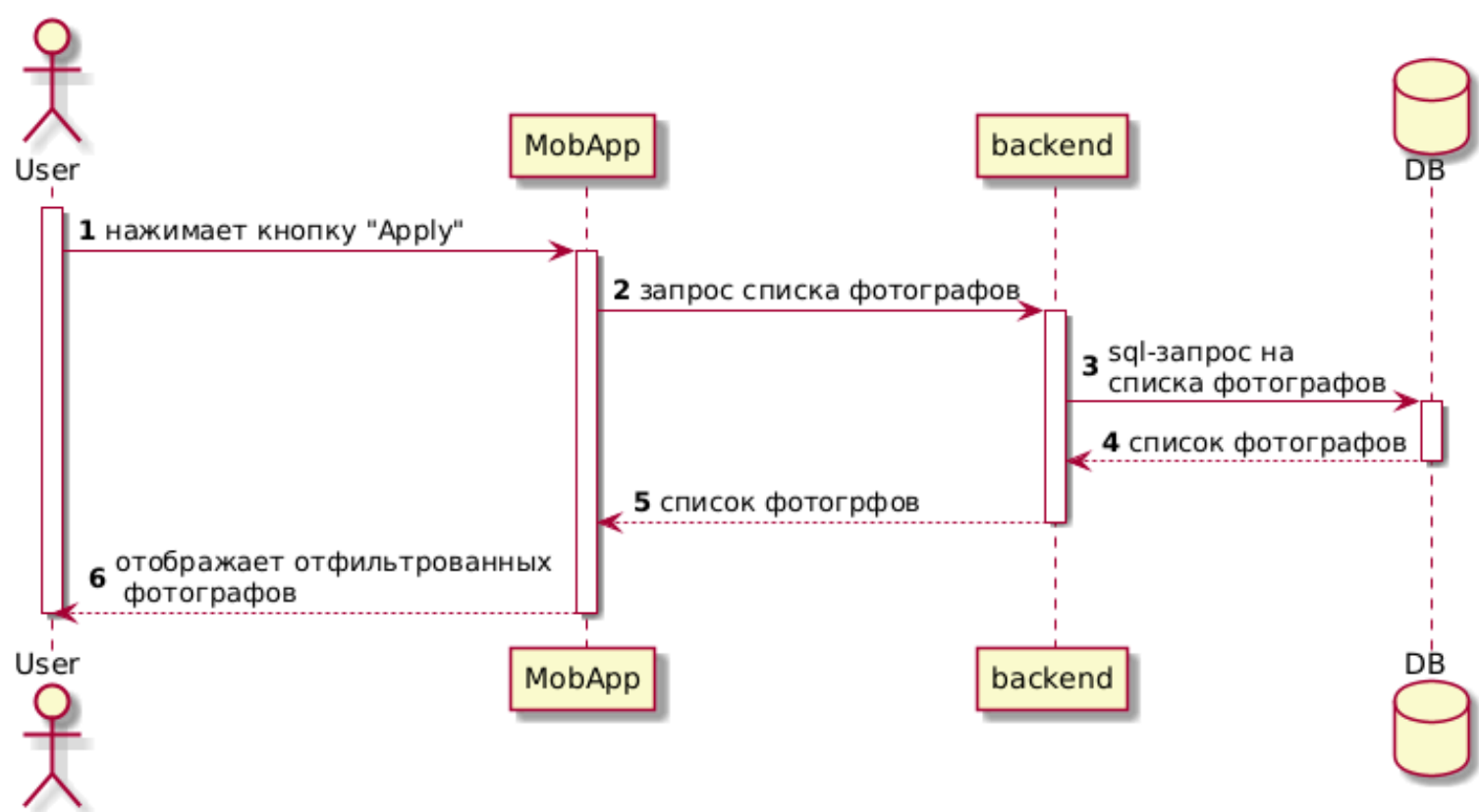
3. ERD-диаграмма

ERD-диаграмма — диаграмма где показано, как разные «сущности» (люди, объекты, концепции и так далее) связаны между собой внутри системы.



## 4. Sequence diagramm (диаграмма последовательностей)

Диаграмма последовательности — это тип UML-диаграммы, который используется для моделирования взаимодействий между объектами в системе. Она отображает, в каком порядке и как взаимодействуют между собой различные объекты или компоненты во времени. Основное назначение — показать, как запросы и ответы передаются между элементами системы.



№	Описание
1.	Пользователь принимает все введенные параметры фильтрации, нажимая кнопку "Apply"
2.	Мобильное приложение направляет запрос поиска фотографий с параметрами фильтрации по локации
3.	Бэк направляет sql запрос на получение списка фотографий
4.	СУБД выполняет запрос и возвращает список фотографий, соответствующих критериям пользователя
5.	Бэк формирует json со списком фотографий и отправляет в приложение
6.	Приложение отображает список отфильтрованных фотографий