

Министерство образования и науки
федеральное государственное автономное образовательное учреждение
высшего образования
«Национальный исследовательский университет ИТМО»
Факультет инфокоммуникационных технологий

Отчет по курсу: «Базы данных»
«Интернет площадка креативных товаров»

Выполнили:

Швалов Даниил Андреевич
Осыченко Арсений Дмитриевич
Богданов Михаил Александрович
Ларичева Дарья Кирилловна
Карагодин Максим Александрович

Группа: K32211

Проверила:

Осетрова Ирина Станиславовна

Санкт-Петербург

2022

Содержание

1. Введение	3
2. Цели	3
3. Задачи	3
4. Концепт интернет-площадки	4
5. Пользовательские возможности	5
5.1 Просмотр и поиск товаров	6
5.2 Взаимодействие с товарами	6
5.3 Вход в аккаунт	10
5.4 Приобретение подписки	12
5.5 Добавление отзывов	14
6. Проектирование базы данных	14
6.1 Пользователи	15
6.2 Товары	20
6.3 Чат между покупателем и продавцом	24
7. Выводы	27

1. Введение

Уже достаточно долгое время интернет-площадки для покупки и продажи вещей пользуются огромной популярностью. На рынке существует множество площадок, предоставляющих сервис для продажи различных видов товаров.

«Креатистор» – это интернет-площадка для креативных людей, интересных идей и уникальных товаров. На рынке существует множество различных магазинов креативных товаров. Однако мы готовы предложить сервис, отличный от существующих.

Мы, равно как наши конкуренты, не покупаем и не продаем товары. Наша компания предоставляем площадку, на которой продавцы находят своих покупателей, а покупатели получают возможность приобрести креативные товары. Однако, в отличие от остальных площадок мы ориентированы не на масс-маркет, а на нишу людей, интересующимся креативными вещами.

2. Цели

Целью данной лабораторной работы является разработка логической схемы базы данных для интернет-площадки по продаже креативных вещей.

3. Задачи

Для достижения поставленной цели необходимо решить следующие задачи:

- анализ предметной области;
- анализ пользовательских возможностей;
- выделение основных сущностей;
- определение связей между сущностями;

- создание UML-модели.

4. Концепт интернет-площадки

Интернет площадка «Креатистор» достаточно сильно похожа на существующие аналоги. Это не удивительно, поскольку достаточно сложно придумать что-то новое в хорошо освоенной сфере. Однако у нашей площадки есть некоторые особенности, которые отличают нас от остальных. Основная особенность заключается в том, что наша площадка ориентирована на креативных людей, предлагающих креативные товары.

Разберем эту идею подробнее, а точнее, каким образом мы собираемся ее реализовать. В первую очередь рассмотрим то, каким способом мы собираемся оставаться площадкой с креативными товарами. Наш способ – модерация. Именно модерация – это первое, что отличает по части реализации нашу платформу от своих аналогов. Поскольку наша площадка стремится быть местом только с креативными товарами, нам необходимо проверять каждое объявление. Это позволит избежать продажи некачественного товара, а также товаров широкого потребления.

Еще одно отличие – это поддержания почти полного равноправия на площадке. Во многих сервисах по продаже товаров предусмотрены способы по продвижению своих товаров. Так, например, это может быть поднятие объявления в поиске или закрепление его. На нашей площадке этого не предусмотрено. Мы считаем, что объявления должны выдаваться только в порядке качества заполнения информации о товаре: чем подробнее, точнее и качественнее написано описание товара, тем более вероятно, что именно этот товар попадет в поиске. В то же время, благодаря модерации будут исключены случаи подкрутки описания под поисковые алгоритмы.

Однако тут же возникает вопрос о монетизации платформы. Именно из-за монетизации на нашей платформе только почти равноправие. Особенностью

нашей платформы является то, что только пользователи с подпиской могут добавлять свои товары. Другими словами, чтобы добавить свой товар на нашу платформу, пользователю необходимо приобрести подписки. Однако это никак не распространяется на просмотр и поиск объявлений: даже незарегистрированный пользователь может искать и просматривать объявления на нашей платформе.

Итого, наша платформа имеет следующие аспекты, отличающие нас от наших конкурентов:

- модерация товаров;
- отсутствие методов продвижения товаров;
- подписочная система.

5. Пользовательские возможности

В этом разделе описан основной функционал, предоставленный пользователю. При взаимодействии с площадкой пользователь может:

- создавать, входить в личный аккаунт;
- просматривать и искать товары;
- приобретать подписку, расширяющую доступный функционал;
- создавать и редактировать товары;
- добавлять товары в избранное;
- оставлять отзывы на продавцов;
- общаться с продавцом с помощью чата.

Рассмотрим каждый функционал поподробнее.

5.1. Просмотр и поиск товаров

Первое, с чем будет взаимодействовать пользователь – это товары, который он присматривает себе. Кроме привычной ленты товаров, наша платформа предлагает поиск по товарам. На нем сделан особый акцент, потому что он несколько отличается от привычного поиска на торговых площадках.

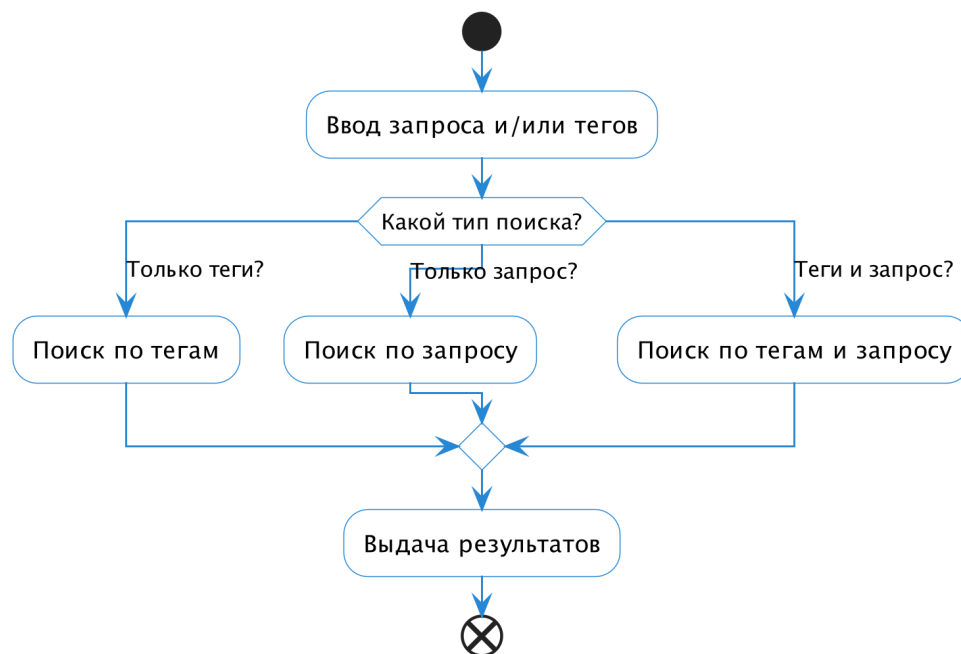


Рисунок 1 – Диаграмма поиска товаров

Как говорилось ранее, наша площадка рассчитана на креативные товары, а потому какие-то конкретные категории выделить очень сложно. Именно поэтому вместо категорий на нашей площадке используются теги. Эти теги похожи на привычные теги из социальных сетей: добавляет их продавец, а нужны они для упрощения поиска. Кроме тегов, также можно использовать обычный текстовый поиск.

5.2. Взаимодействие с товарами

Основной функционал нашей площадки завязан на товарах. В этом разделе будет рассмотрены следующие возможности по взаимодействию с товарами:

- выставление товара на продажу;
- изменение товара;
- добавление товара в избранное.

Выставление товара на продажу

Рассмотрим подробнее процесс выставления товару на продажу. Для этого зарегистрированный пользователь с подпиской должен заполнить форму, в которой он должен добавить следующую информацию о товаре:

- название товара;
- стоимость;
- описание;
- изображения товара.

Кроме того, пользователь может добавить свои контактные данные в виде телефона и адреса электронной почты.

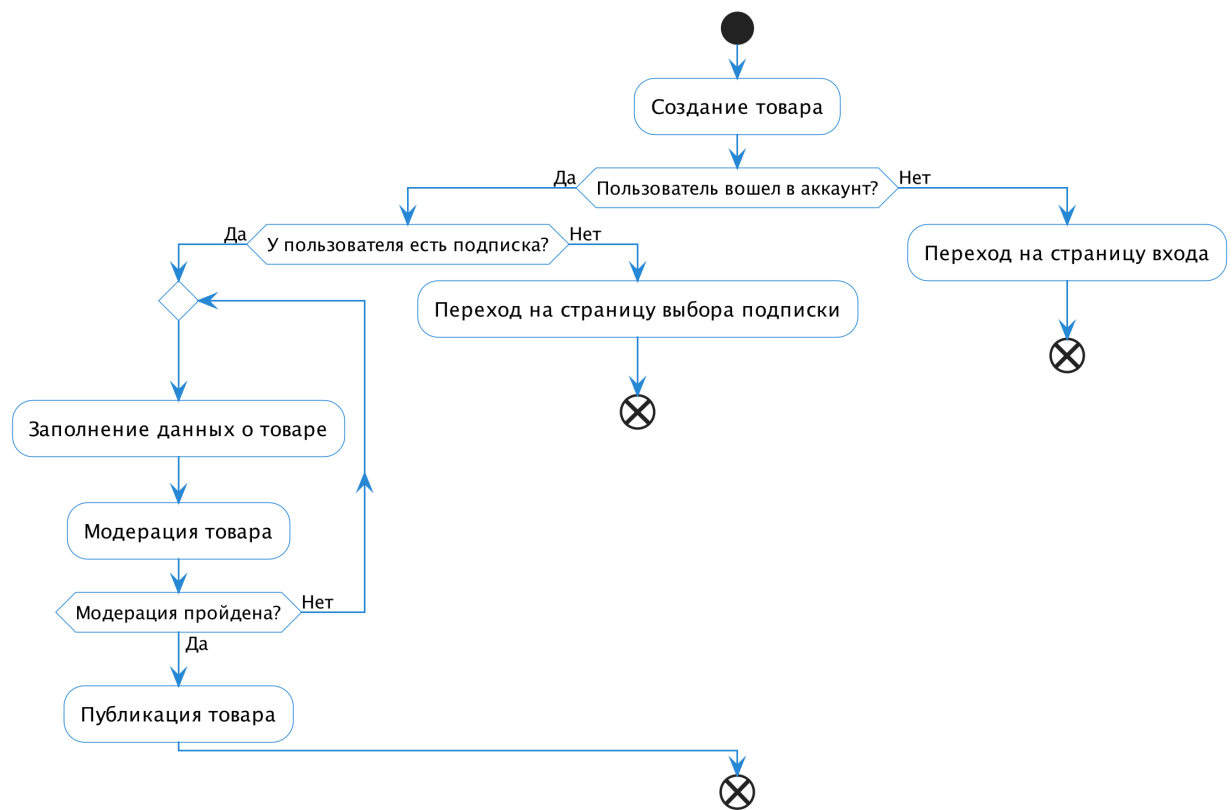


Рисунок 2 – Диаграмма добавления товара на продажу

После заполнения формы товар отправляется на модерацию. На модерации товар проверяется по следующим критериям:

- *уникальность*: товар не должен быть самым обычным, доступным на каждом углу;
- *приемлемое качество*: товар не должен выглядеть как мусор;
- *креативность*: товар должен быть интересным, цепляющимся за глаз.

Таким образом, модерацию не смогут пройти следующие вещи:

- товары широкого потребления;
- товары низкого качества;
- товары, не содержащие креативной идеи.

Изменение товара

После выставления товара на продажу, через некоторое время может появиться необходимость в изменении состояния товара. Так, пользователю доступны следующие состояния товара:

- доступен к покупке;
- нет в наличии;
- продан.

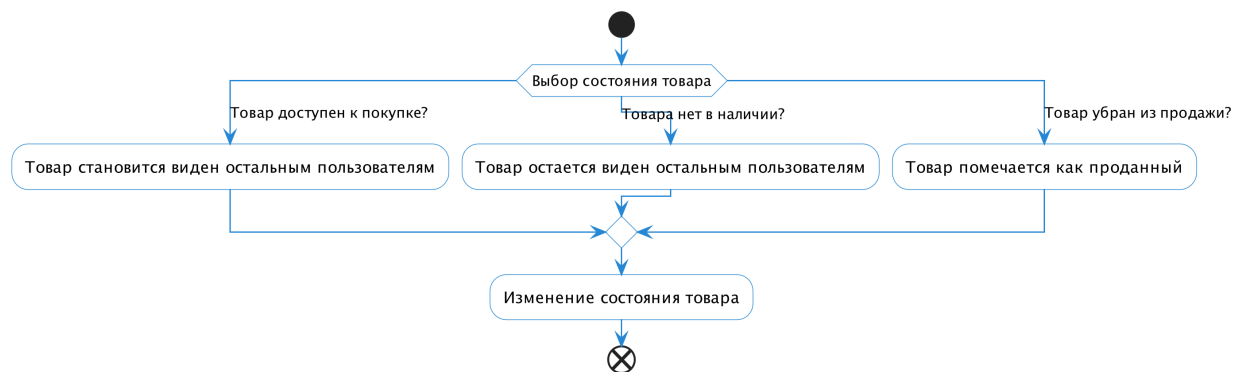


Рисунок 3 – Диаграмма изменения состояния товара

В случае, когда товар помечается как доступный к покупке, товар становится виден всем пользователям площадки. При изменении состояния на нет в наличии, товар все еще остается виден пользователям. Однако, если пользователь пометит товар как проданный, то товар будет виден только продавцу. Остальные пользователи площадки не смогут найти его.

Добавление товара в избранное

Чаще всего пользователи делают выбор из нескольких товаров. Поэтому было бы удобно, если можно было бы сохранить понравившиеся товары в отдельный список. Эту функцию реализует система избранных товаров.

Пользователю достаточно всего лишь добавить товар в избранное, чтобы не потерять его.

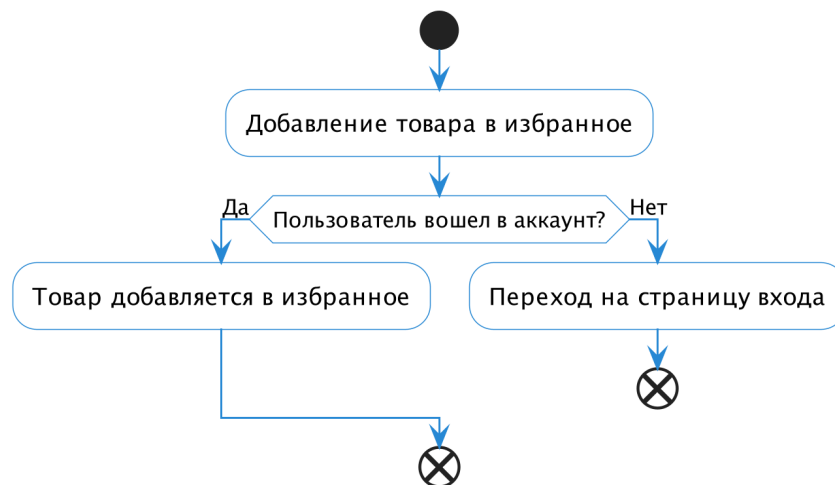


Рисунок 4 – Диаграмма добавления товара в избранное

Единственное условие – пользователь должен быть авторизован. Мы требуем это, поскольку товары, добавленные в избранное, относятся к конкретному пользователю.

5.3. Вход в аккаунт

Теперь рассмотрим процесс входа в аккаунт. Как и всегда, вход в аккаунт делится на две составляющие: регистрацию и авторизацию.

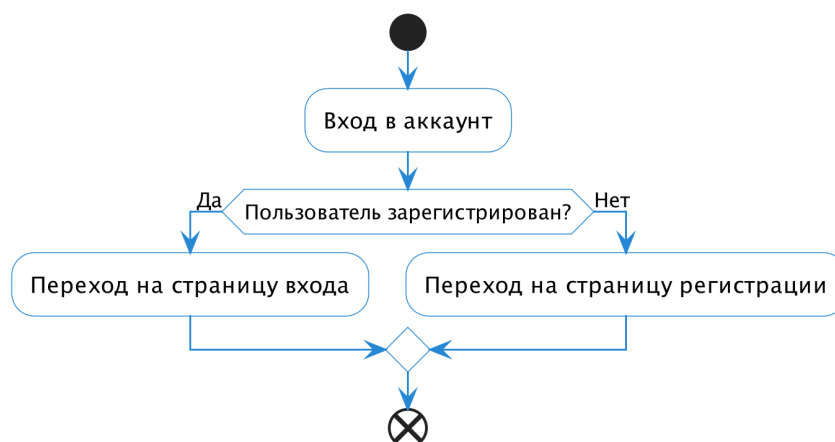


Рисунок 5 – Диаграмма входа в аккаунт

Регистрация

При регистрации проверяется корректность введенных данных. Точнее, проверяются следующие критерии:

- пользователя с данной электронной почтой не существует;
- пользователя с данным ником не существует;
- ник не содержит запрещенных слов;
- пароль состоит минимум из 8 символов.

После регистрации пользователь попадает на страницу авторизации.

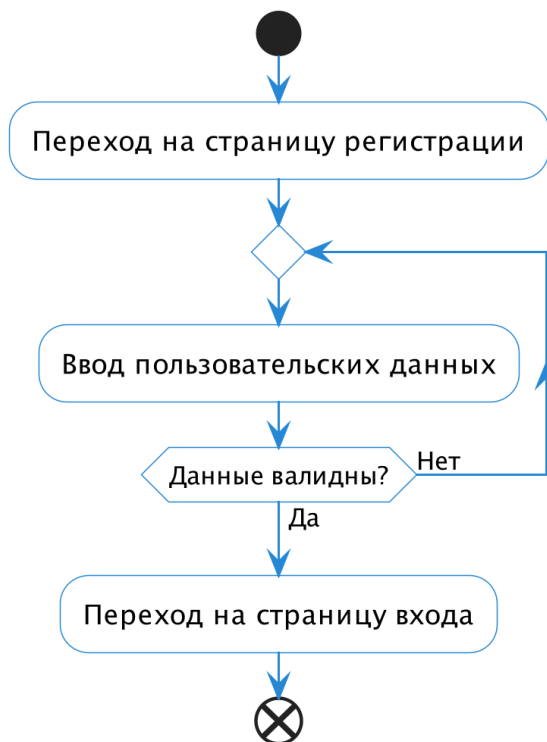


Рисунок 6 – Диаграмма регистрации

Авторизация

При авторизации сначала проверяется корректность введенных данных. После этого, проверяется, подтвержден ли текущий аккаунт. Аккаунт под-

тверждается с помощью перехода по ссылке, которая приходит на электронную почту. Если аккаунт подтвержден, то процесс авторизации завершен. Иначе, пользователь может потребовать выслать новую ссылку для подтверждения аккаунта.

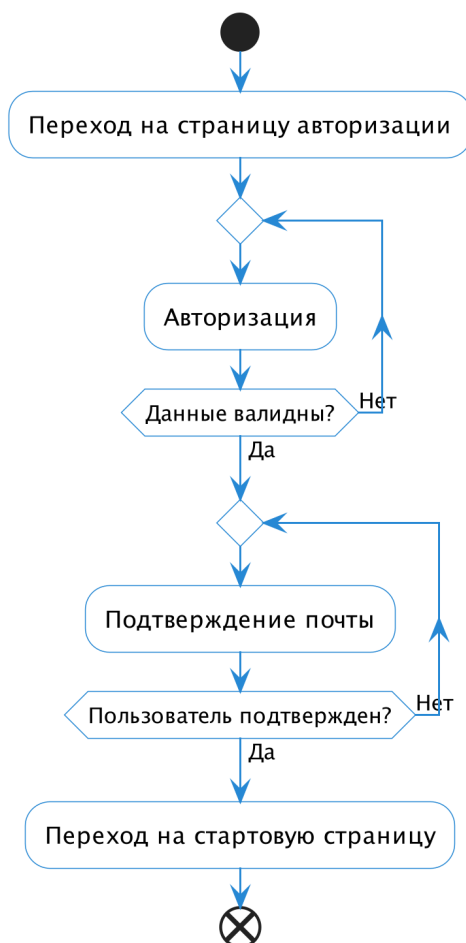


Рисунок 7 – Диаграмма авторизации

5.4. Приобретение подписки

Для добавления товара на площадку необходимо выполнить как минимум два условия: быть авторизованным и иметь подписку на своем аккаунте. Рассмотрим процесс приобретения подписки подробнее.

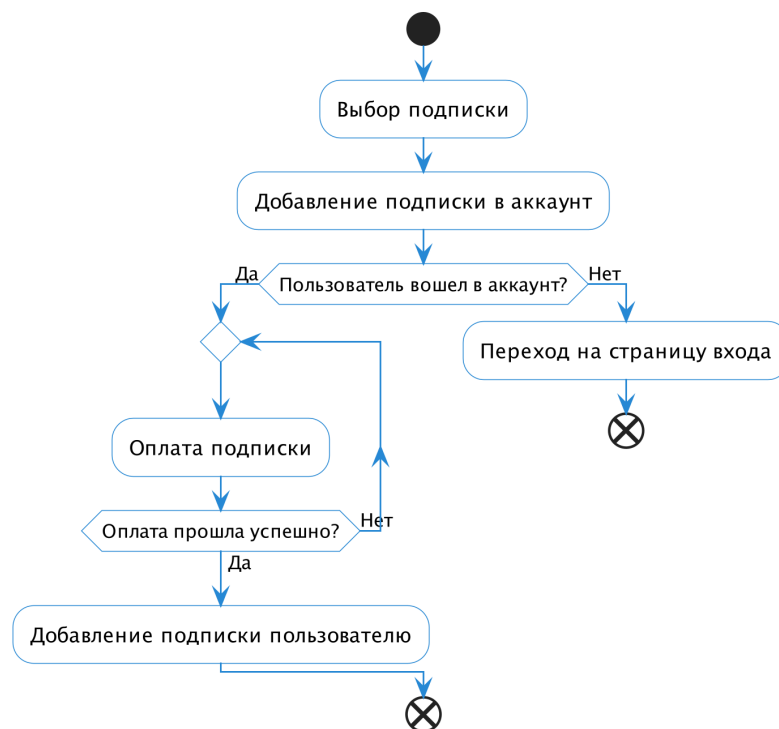


Рисунок 8 – Диаграмма добавления подписки

Прежде всего, пользователю необходимо определиться, какую подписку он собирается приобрести. В следующей таблице приведены примерные виды подписки.

Таблица 1 – Примеры подписок

Название	Длительность	Стоимость
Право имеющий	1 месяц	300 рублей
Твоя роза	3 месяца	777 рублей
Замена счастью	6 месяцев	1337 рублей
Часть той силы	12 месяцев	2048 рублей

После выбора подписки, пользователь должен оплатить ее. В случае успешного завершения оплаты, подписка добавляется на аккаунт пользователя.

5.5. Добавление отзывов

Пользователи могут добавлять отзывы на продавцов. В отличие от товаров, отзывы видны сразу после их добавления. Однако отзыв может быть удален, если он не пройдет модерацию.

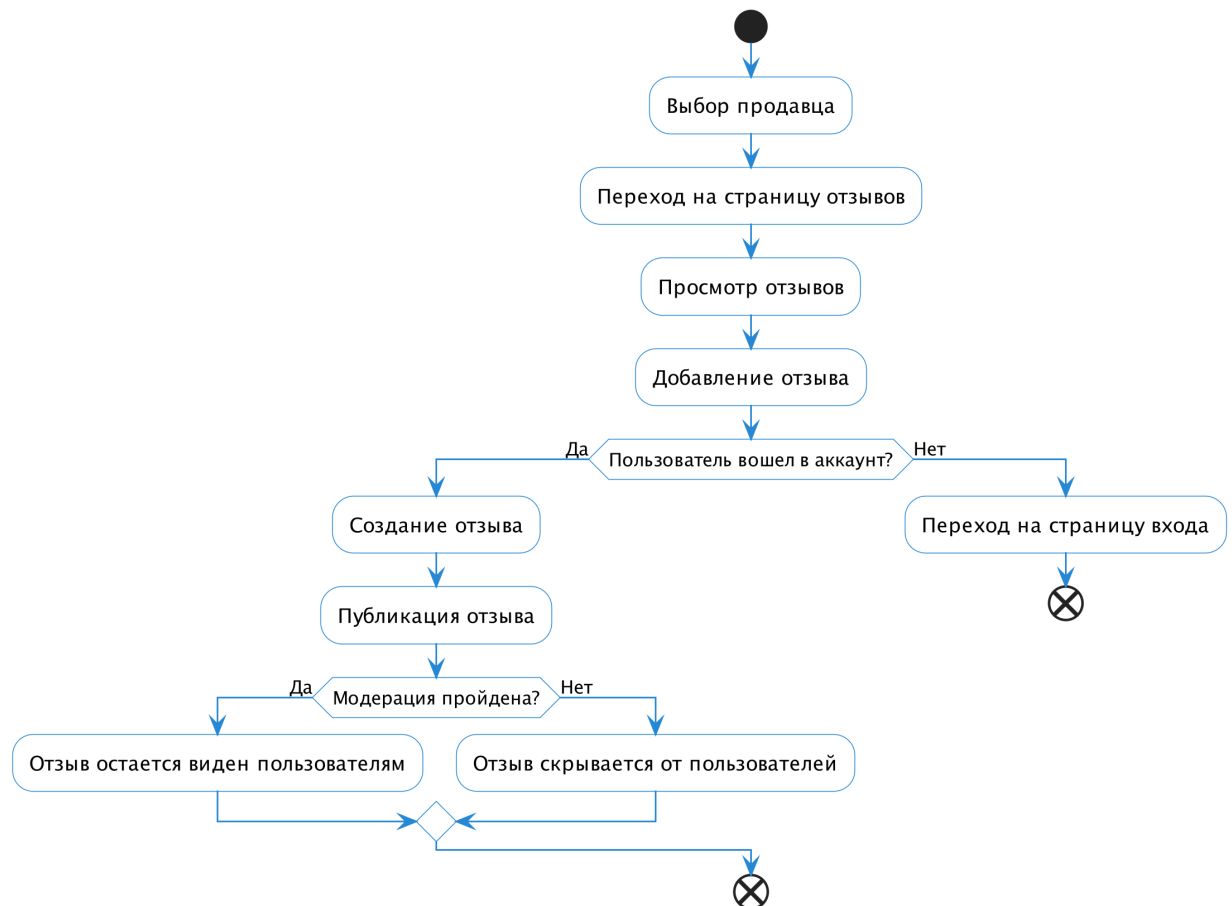


Рисунок 9 – Диаграмма добавления отзыва к продавцу

6. Проектирование базы данных

На рисунке 10 представлена схема получившейся в ходе разработки базы данных. В этом разделе мы подробно рассмотрим причины тех или иных решений.

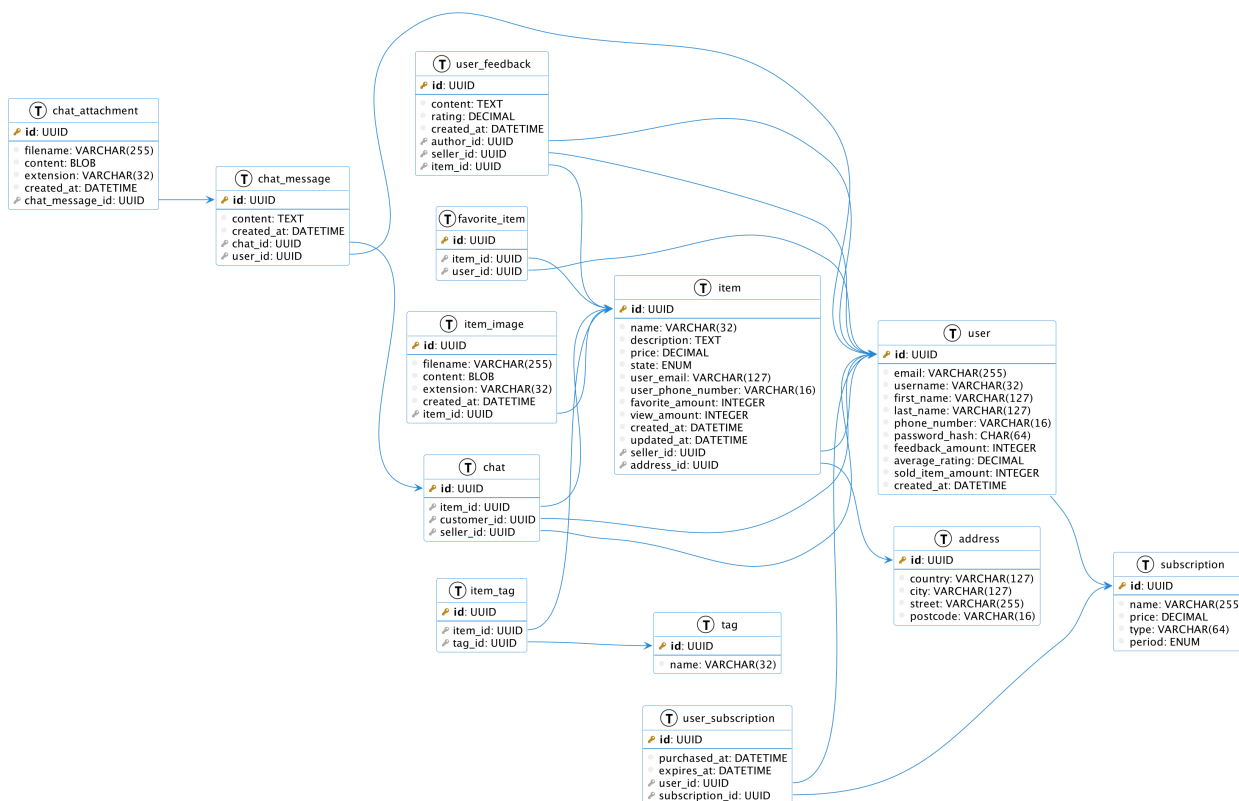


Рисунок 10 – Диаграмма базы данных

6.1. Пользователи

Для начала рассмотрим устройство пользователя в нашей системе. Прежде всего каждый пользователь должен иметь информацию об электронной почте, нике и пароле. Для безопасности пароль будем хранить в захешированном виде. Кроме того, пользователь также может указать информацию о своем имени и фамилии, номере телефона. Поскольку мы предполагаем наличие рейтинговой системы у каждого пользователя, нам также нужно хранить количество отзывов и их среднее значение. Также было решено хранить дату создания аккаунта и количество проданных товаров. Описание получившихся полей этой таблицы представлено в таблице 2.

Таблица 2 – Описание полей таблицы user

Название	Тип	Описание
----------	-----	----------

id	UUID	Идентификатор пользователя. Первичный ключ
email	VARCHAR(255)	Почта пользователя. Обязательное поле
username	VARCHAR(32)	Ник пользователя. Обязательное поле
first_name	VARCHAR(127)	Имя пользователя
last_name	VARCHAR(127)	Фамилия пользователя
phone_number	VARCHAR(16)	Телефонный номер пользователя
password_hash	CHAR(16)	Хеш пароль пользователя (SHA-256). Обязательное поле
feedback_amount	INTEGER	Количество отзывов у пользователя как продавца. Значение по умолчанию – 0
average_rating	DECIMAL	Средний рейтинг пользователя как продавца. Значение варьируется в интервале от 1 до 5
sold_item_amount	INTEGER	Количество проданных товаров. Значение по умолчанию – 0
created_at	DATETIME	Дата создания аккаунта

Теперь рассмотрим подписочную систему. Прежде всего, у подписки должны быть определены названия, стоимость и время действия. Кроме того, в каждой подписке будем хранить ее тип. Это позволит гибко фильтровать подписки в зависимости от ситуации (например, нам нужно ввести подписки для Нового года, при этом должна быть возможность отображать как старую, так и новую цену). Описание получившихся полей этой таблицы представле-

но в таблице 3.

Таблица 3 – Описание полей таблицы subscription

Название	Тип	Описание
id	UUID	Идентификатор подписки. Первичный ключ
name	VARCHAR(255)	Название подписки. Обязательно поле
price	DECIMAL	Стоимость подписки. Обязательное поле
type	VARCHAR(64)	Тип подписки. Обязательное поле
period	ENUM	Время действия подписки. В ENUM содержатся следующие значения: MONTH, THREE MONTH, HALF YEAR, YEAR

Теперь рассмотрим то, как связаны пользователи и подписки. Несмотря на то, что в конкретный момент времени пользователь может иметь только одну активную подписку, нам необходимо хранить не только текущую, но и все предыдущие подписки. Это позволит, при необходимости, ввести систему скидок, которая будет зависеть от количества предыдущих купленных подписок.

Каждая подписка пользователя должна содержать информацию о начале и об окончании срока подписки, а также идентификаторы пользователя и конкретной подписки. Исходя из этого описания, получаем, что мы имеем связь многие ко многим, поскольку один пользователь мог использовать несколько разных подписок за все время, равно как и одна подписка может использоваться у нескольких пользователей сразу. Описание получившихся полей этой таблицы представлено в таблице 4.

Таблица 4 – Описание полей таблицы user_subscription

Название	Тип	Описание
id	UUID	Идентификатор подписки пользователя. Первичный ключ
purchased_at	DATETIME	Дата приобретения подписки. Обязательное поле
expires_at	DATETIME	Дата истечения подписки. Обязательное поле
user_id	UUID	Идентификатор пользователя. Внешний ключ на таблицу user поле id
subscription_id	UUID	Идентификатор подписки. Внешний ключ на таблицу subscription поле id

Рассмотрим устройство отзывов на продавцов. Каждый отзыв должен включать в себя свое содержимое, оценку продавца и дату создания отзыва. Кроме этого, поскольку отзывы завязаны на определенный товар, продавца и покупателя, мы должны хранить идентификаторы этих сущностей. Описание получившихся полей этой таблицы представлено в таблице 5.

Таблица 5 – Описание полей таблицы user_feedback

Название	Тип	Описание
id	UUID	Идентификатор отзыва. Первичный ключ
content	TEXT	Текстовое содержание отзыва. Обязательное поле

rating	TEXT	Оценка, которую поставил пользователь продавцу. Обязательное поле. Значение варьируется в интервале от 1 до 5
created_at	DATETIME	Дата создания отзыва. Обязательное поле
author_id	UUID	Идентификатор автора отзыва. Внешний ключ на таблицу user поле id
seller_id	UUID	Идентификатор продавца. Внешний ключ на таблицу user поле id
item_id	UUID	Идентификатор товара. Внешний ключ на таблицу item поле id

Рассмотрим систему добавления товара в избранное. Для реализации этой системы нам необходимо хранить идентификатор пользователя, добавившего товар, и идентификатор самого товара. Из этого следует, что имеем связь многие ко многим, поскольку несколько товаров может быть добавлено в избранное одним пользователем. В то же время один товар может быть добавлен в избранное несколькими пользователями. Описание получившихся полей этой таблицы представлено в таблице 6.

Таблица 6 – Описание полей таблицы favourite_item

Название	Тип	Описание
id	UUID	Идентификатор понравившегося товара. Первичный ключ
item_id	UUID	Идентификатор товара. Внешний ключ на таблицу item поле id

user_id	UUID	Идентификатор пользователя. Внешний ключ на таблицу user поле id
---------	------	--

6.2. Товары

Рассмотрим устройство товаров. Каждый товар должен включать название, описание и стоимость. Кроме того, он дополнительно может включать в себя информацию о электронной почте и телефонном номере продавца. Также у каждого товара есть состояние (например, товар находится на модерации). Также, поскольку у пользователей есть возможность добавлять товары в избранное, было решено добавить счетчик людей, добавивших в избранное. Чтобы лучше продвигать тот или иной товар, было решено также хранить количество просмотров товара. Кроме того, мы решили добавить дату создания и дату изменения товара. В добавок ко всему вышеперечисленному, нам необходимо хранить идентификатор продавца. Также было решено у каждого объявления хранить адрес, чтобы ранжировать объявления по городам.

Исходя из вышеперечисленной информации, мы получаем связь один ко многим. Все потому, что у товара может быть только один продавец и один адрес. В это же время один и тот же адрес может быть у нескольких товаров, равно как и продавец может предлагать несколько товаров. Описание получившихся полей этой таблицы представлено в таблице 7.

Таблица 7 – Описание полей таблицы item

Название	Тип	Описание
id	UUID	Идентификатор товара. Первичный ключ

name	VARCHAR(32)	Название товара. Обязательное поле
description	TEXT	Описание товара. Обязательное поле
price	DECIMAL	Стоимость товара. Обязательное поле
state	ENUM	Состояние товара. Обязательное поле. В ENUM содержатся следующие значения: ON MODERATION, IN STOCK, OUT OF STOCK, SOLD, REMOVED
user_email	VARCHAR(127)	Электронная почта пользователя
user_phone_number	VARCHAR(16)	Телефонный номер пользователя
favourite_amount	INTEGER	Количество пользователей, добавивших товар в избранное. Значение по умолчанию – 0
view_amount	INTEGER	Количество просмотров товара. Значение по умолчанию – 0
created_at	DATETIME	Дата создания товара. Обязательное поле
updated_at	DATETIME	Дата изменения товара
seller_id	UUID	Идентификатор продавца. Внешний ключ на таблицу user поле id
address_id	UUID	Адрес продажи товара. Внешний ключ на таблицу address поле id

Рассмотрим систему картинок товара. К каждому товару может быть приложено несколько картинок. Каждая картинка должна включать в себя имя файла, содержимое и расширение. Кроме этого картинка привязана к определенному товару. Исходя из этого получаем связь один ко многим, поскольку картинка может относиться только к одному товару. Однако в то же время у одного товара может быть несколько картинок. Описание получившихся полей этой таблицы представлено в таблице 8.

Таблица 8 – Описание полей таблицы `item_image`

Название	Тип	Описание
id	UUID	Идентификатор изображения товара. Первичный ключ
filename	BLOB	Имя файла. Обязательное поле
content	BLOB	Двоичные данные файла. Обязательное поле
created_at	DATETIME	Дата создания изображения
item_id	UUID	Идентификатор товара. Внешний ключ на таблицу <code>item</code> поле <code>id</code>

Рассмотрим систему тегов. Она довольно проста, поскольку тег должен включать в себя только имя. Описание получившихся полей этой таблицы представлено в таблице 9.

Таблица 9 – Описание полей таблицы `tag`

Название	Тип	Описание
id	UUID	Идентификатор тега. Первичный ключ
name	VARCHAR(32)	Название тега. Обязательное поле

Теперь рассмотрим применение тегов с товарами. Для этого необходимо хранить идентификатор товара, к которому прикреплен тег, а также идентификатор самого тега. Значит, мы имеем дело с связью многие ко многим, поскольку один товар может иметь несколько тегов, а один тег может относиться к нескольким товарам. Описание получившихся полей этой таблицы представлено в таблице 10.

Таблица 10 – Описание полей таблицы `item_tag`

Название	Тип	Описание
<code>id</code>	UUID	Идентификатор тега товара. Первичный ключ
<code>item_id</code>	UUID	Идентификатор товара. Внешний ключ на таблицу <code>item</code> поле <code>id</code>
<code>tag_id</code>	UUID	Идентификатор тега. Внешний ключ на таблицу <code>tag</code> поле <code>id</code>

Рассмотрим адрес, используемый в описании товара. Каждый адрес обязательно должен включать в себя информацию о стране, городе и улице. Кроме этой информации, адрес может включать в себя почтовый индекс. Описание получившихся полей этой таблицы представлено в таблице 11.

Таблица 11 – Описание полей таблицы `address`

Название	Тип	Описание
<code>id</code>	UUID	Идентификатор адреса. Первичный ключ
<code>country</code>	VARCHAR(127)	Название страны. Обязательное поле
<code>city</code>	VARCHAR(127)	Название города. Обязательное поле
<code>street</code>	VARCHAR(127)	Название улицы. Обязательное поле

postcode	VARCHAR(16)	Почтовый индекс
----------	-------------	-----------------

6.3. Чат между покупателем и продавцом

Перед покупкой товара у пользователя может появиться потребность пообщаться с продавцом. Например, покупатель захотел разузнать о товаре подробнее или, например, пользователь готов купить товар. Для этого покупателю необходимо как-нибудь связаться с продавцом. Наша площадка предлагает несколько путей для решения этой проблемы.

Как было показано ранее, у каждого товара обязательно есть описание. Кроме того, к описанию можно добавить электронную почту и телефонный номер продавца. Таким образом, с помощью каких-либо контактных данных в описании товара, электронной почты или телефона покупатель может связаться с продавцом для уточнения своих вопросов.

Однако это не единственный способ связи покупателя с продавцом. Наша площадка предоставляет минимально необходимый функционал для общения покупателей и продавцов. Для каждого товара между покупателем и продавцом создается чат, в котором можно отправлять как текстовые сообщения, так и различные файлы, в том числе и картинки.

Исходя из вышеописанного функционала, можно выделить три сущности: чат, сообщения и вложения. Рассмотрим каждый из них поподробнее.

Начнем с таблицы чатов. Как было сказано ранее, чаты формируются отдельно для покупателя и продавца для каждого товара. Значит, нам нужно хранить идентификаторы покупателя и продавца, а также идентификатор товара.

Таким образом, мы получаем связь один к многим, поскольку конкретный чат может быть только между определенными пользователями. В то же время

одни и те же покупатель и продавец могут переписываться на счет разных товаров. При этом, один продавец может иметь чат с несколькими клиентами для одного товара, а клиент может переписываться с несколькими продавцами на счет нескольких товаров сразу. Описание получившихся полей этой таблицы представлено в таблице 12.

Таблица 12 – Описание полей таблицы chat

Название	Тип	Описание
id	UUID	Идентификатор чата. Первичный ключ
item_id	UUID	Идентификатор товара. Внешний ключ на таблицу item поле id
customer_id	UUID	Идентификатор покупателя. Внешний ключ на таблицу user поле id
seller_id	UUID	Идентификатор продавца. Внешний ключ на таблицу user поле id

Теперь рассмотрим таблицу, отвечающую за хранение конкретного сообщения. Прежде всего каждое сообщение отправляется определенным пользователем, поэтому нам необходимо хранить идентификатор автора сообщения. Кроме того, каждое сообщение привязано к какому-то чату, значит, мы также должны хранить идентификатор чата. Кроме идентификаторов, также необходимо хранить содержимое сообщения и дату создания сообщения.

Исходя из описания, мы получаем связь один ко многим, поскольку определенное сообщение может быть отправлено только одним пользователем в определенном чате. В то же время, пользователь может отправлять несколько сообщений в разных чатах. Описание получившихся полей этой таблицы представлено в таблице 13.

Таблица 13 – Описание полей таблицы chat_message

Название	Тип	Описание
id	UUID	Идентификатор сообщения. Первичный ключ
content	TEXT	Содержание сообщения. Обязательное поле
created_at	DATETIME	Дата создания сообщения. Обязательное поле
chat_id	UUID	Идентификатор чата. Внешний ключ на таблицу chat поле id
user_id	UUID	Идентификатор автора сообщения. Внешний ключ на таблицу user поле id

Рассмотрим таблицу, которая будет хранить в себе содержимое вложения. В первую очередь нам необходимо хранить идентификатор сообщения, к которому относится это вложение. Кроме него, мы также будем хранить имя приложенного файла, его содержимое и расширение этого файла. Также будем хранить дату создания этого вложения.

В этот раз мы также получили связь один ко многим, потому что одно сообщение может включать в себя несколько вложений, однако одно и то же вложение может относиться только к одному сообщению. Описание получившихся полей этой таблицы представлено в таблице 14.

Таблица 14 – Описание полей таблицы chat_attachment

Название	Тип	Описание
id	UUID	Идентификатор вложения. Первичный ключ
filename	VARCHAR(255)	Имя файла. Обязательное поле

content	BLOB	Двоичные данные файла. Обязательное поле
extension	VARCHAR(32)	Расширение файла. Обязательное поле
created_at	DATETIME	Дата загрузки файла. Обязательное поле
chat_message_id	UUID	Идентификатор сообщения. Внешний ключ на таблицу chat_message поле id

7. Выводы

В результате выполнения лабораторной работы удалось спроектировать базу данных для интернет площадки для креативных людей. В процессе выполнения работы были выполнены все поставленные задачи.

Однако во время выполнения возникли некоторые сложности. В первую очередь, было очень сложно сбалансировать систему таким образом, чтобы пользователь чувствовал себя наравне с другими пользователями. При этом, проект нужно каким-то образом монетизировать. В качестве решения этой проблемы была выбрана подписочная система.

Еще одной сложностью стала проблема проектирования. Необходимо было спроектировать систему так, чтобы база данных была готова как к расширению, так и к более серьезным нагрузкам.

В итоге все проявившиеся проблемы удалось решить, как с точки зрения концепта, так и с точки зрения архитектуры базы данных.